

國立暨南國際大學資訊工程學系
碩士論文

開發手繪為基礎的 3D NPR 水彩繪畫系統圖
形使用者介面

**Developing a Sketch-Based GUI of 3D
Watercolor NPR System**

指導教授：陳履恆博士

研究生：何宜欣

中華民國 九十八 年 六 月

國立暨南國際大學碩（博）士論文考試審定書

資訊工程 學系（研究所）

研究生 何宜欣 所提之論文

開發以手繪為基礎的 NPR 水彩繪畫系統圖形使用者介面
Developing a Sketch-Based GUI of 3D Watercolor NPR System
(中、英文題目)

經本委員會審查，符合碩（博）士學位論文標準。

學位考試委員會

張鈞法

委員兼召集人

歐陽明

委員

陳復恆

委員

中華民國 98 年 6 月 26 日

博碩士論文電子檔案上網授權書

(提供授權人裝訂於紙本論文書名頁之次頁用)

本授權書所授權之論文為授權人在 暨南國際大學 資訊工程學系
97 學年度第 二 學期取得 碩士 學位之論文。

論文題目：

開發手繪為基礎的 3D NPR 水彩繪畫系統圖形使用者介面

指導教授： 陳履恆

茲同意將授權人擁有著作權之上列論文全文（含摘要），非專屬、無償授權國家圖書館及本人畢業學校圖書館，不限地域、時間與次數，以微縮、光碟或其他各種數位化方式將上列論文重製，並得將數位化之上列論文及論文電子檔以上載網路方式，提供讀者基於個人非營利性質之線上檢索、閱覽、下載或列印。

- 讀者基非營利性質之線上檢索、閱覽、下載或列印上列論文，應依著作權法相關規定辦理。

授權人：何宜欣

簽 名： 何宜欣

中華民國 98 年 07 月 16 日

國立暨南國際大學博碩士論文授權書

本授權書所授權之論文為立書人在 國立暨南國際大學科技學院 資訊工程學系，97學年度第2學期取得碩士學位之論文。

論文題目：開發手繪為基礎的3D NPR水彩繪畫系統圖形使用者介面

指導教授：陳履恆

授權事項：

- 一、立書人 何宜欣 同意無償授權 國立暨南國際大學 將上列論文全文資料之以微縮、數位化或其他方式進行重製作為典藏之用。國立暨南國際大學 在上述範圍內得再授權第三人進行重製。
- 二、立書人何宜欣 同意有償授權 將前條典藏之資料收錄於資料庫，並以電子形式透過單機、網際網路、無線網路或其他傳輸方式授權用戶進行檢索、瀏覽、下載、傳輸、列印等。國立暨南國際大學得將上述權利再授權于第三者。
- 前兩條授權均為非專屬授權，立書人仍擁有上述授權著作之著作權。立書人擔保本著作為立書人所創作之著作，有權依本授權書內容進行各項授權，且未侵害任何第三人之智慧財產權。如有侵害他人權益及觸犯法律之情事，立書人願自行負責一切法律責任，被授權人一概無涉。

有償授權條件：享有權利金的回饋，權利金捐贈校務發展基金指定用於圖書館館務使用。

論文全文上載網路公開時間：立即公開

立書人：何宜欣

簽名：何宜欣

中華民國 98 年 7 月 17 日

致謝

此論文的完成，首先要感謝的，是我的指導教授 陳履恆博士。陳老師給予我研究上的幫助以及指導，並且提供給我充分的獨立思考與發揮機會。在需要協助與鼓勵的時候，給予最強大的支援，讓我得以專注的學習，並且學到了許多方面的知識與解決問題的方法。此外，我也要感謝口試老師 張鈞法博士與 歐陽明博士給予我論文上的指導與建議，使我的論文能夠更加完整與嚴謹。

我也要感謝研究室的岐智、竣蘭學長以及宜靜學姐，給予我各方面的協助與鼓勵，還有沂顯、昱升以及亭瑜同學這兩年來的鼓勵與陪伴，也感謝學弟們的支援與關心。還有要感謝 SRT 的同事們的幫助，以及大學同學凡之、江曼、子熒與佳臻，他們給予我一直以來的關心。

最後，我要感謝我的父母與家人，無論什麼時候，總是給我最多的支持與鼓勵，讓我無論遇到什麼困難，都能夠毫不懼怕地勇敢追尋自己的目標，並也完成了這篇論文。

論文名稱：開發手繪為基礎的 3D NPR 水彩繪畫系統圖形使用者介面

校院系：國立暨南國際大學資訊工程學系

畢業時間：98/6

研究生：何宜欣

頁數：59

學位別：碩士

指導教授：陳履恆 博士

摘要

Non-Photorealistic Rendering(NPR)在圖學的領域中由來已久，此一特殊的成像方式不以物品的擬真度為優先考量，而是著重於筆觸(stroke)、繪畫風格(Paint style)等的呈現。NPR 手法的特徵之一，是以誇張的手法呈現欲突顯的部分，而不注重其呈現出的正確精確性。

3D NPR 系統通常能夠以 3D 模型及場景資訊自動產生繪畫影像，但是產生出來的影像卻難以修改，間接的限制了讓使用者創作的空間。

在本研究計劃中，我們開發一個以手繪為基礎的 NPR 水彩繪畫系統圖形使用者介面。除了能夠讓使用者自由加入不同的 3D 模型及調整場景資訊的水彩 NPR 系統，經由此介面，使用者能夠以較直覺的手繪方式修改原本畫在物體上的色彩。在此介面當中，我們提供一個調色盤。使用者可以藉由輸入的畫作來產生一個可供配色參考的調色盤，或是經由選擇幾個必須的顏色來產生不同色調的調色盤。使用者也能夠運用我們所提供的調色盤在原本的顏色上做些許的變更，並且模擬在 3D 物體上的色彩混合現象，或以互動的方式在背景或是物體上增加許多有趣的視覺效果。

在計算動態的水彩流體方面，我們使用 Lattice Boltzmann Method (LBM) 來模擬水及顏料在紙上的流體作用。經由這個系統，我們以簡單且容易使用的方法將水彩畫家慣用的繪畫技巧生動地模擬出來。

關鍵字：電腦圖學;非擬真成像;晶格波茲曼;水彩;渲染;顏色混色;圖形使用者介面;
手繪

Title of Thesis: Developing a Sketch-Based GUI of 3D Watercolor NPR System

Name of Institute: National Chi Nan University, Dept. of CSIE

Pages:59

Graduation Time: 6/2009

Degree Conferred: Master

Student Name: Yi-Hsin Ho

Advisor Name: Dr. Lieu-Hen Chen

Abstract

Non-Photorealistic Rendering (NPR) has been an important research topic in the field of Computer Graphics. Instead of highlighting the realism of CG synthesized images, NPR focus on the representation of the stroke and painter's style.

3D NPR system is able to render images by the information from the 3D model and the specific scene automatically, but the images are difficult to be modified interactively. The problem indirectly limits the creativity of users.

In our system, we develop a sketch-based GUI for a 3D NPR painting system. The system not only enables users to compose their scene freely with different 3D models, through this interface, user can also manipulate watercolor attributes of object intuitively and manually. In the interface, we offer a color palette. User can create a palette by selecting a painting, or choosing a few needed colors to produce a palette that has a unique color tone. A user can create subtle changes in colors and simulates the blending effects to 3D object through our color palette. Furthermore, user can add many interesting visual effects to the background and on the objects interactively.

The Lattice Boltzmann Method (LBM) is adopted to compute the fluid dynamics of water and pigments on a paper. Through our system, the painting skill of watercolor artists is simulated easily in a user-friendly manner.

Keywords:Computer Graphic(CG);Non-Photorealistic Rendering(NPR);Lattice Boltzmann Method(LBM);Watercolor;Washing;Color Mixing;Graphic User Interface;Sketch-Based

目錄

摘要	II
Abstract	III
目錄	IV
圖目錄	VI
第一章 序論	1
1.1 研究動機與目的	1
1.2 論文編排模式	2
1.3 本論文的貢獻	3
第二章 研究背景	4
2.1 NPR 概述	4
2.2 水彩概述	7
2.2.1 水彩工具與材料	7
2.2.2 水彩技法	8
2.2.3 色彩與配色	10
2.3 水彩畫法的分析與量化	12
2.4 相關研究	14
第三章 Lattice Boltzmann Method	18
3.1 簡介	18
3.2 LBE 的兩個運作步驟	19
第四章 系統概述	21
4.1 系統架構及運作流程	21
4.1.1 系統架構	21
4.1.2 運作流程	22
4.2 資料讀取及虛擬畫布	24
4.2.1 資料讀取及整理	24
4.2.2 虛擬畫布：Modified Z-buffer	26
第五章 水彩模擬	31
5.1 Simulation Overview	31
5.2 水彩模擬流程圖	31
5.3 color blending	37
第六章 GUI 設計	39
6.1 System overview	39
6.2 繪圖板整合	40
6.3 Color Chart 系統整合	41
第七章 筆觸模擬	43

7.1 自動筆觸模擬	43
7.1.1 2D Shape-based Stroke.....	45
7.1.2 3D model-based Stroke	46
7.1.3 Specular	48
7.1.4 描繪過程	49
7.2 手繪筆觸模擬	50
第八章 研究成果	52
第九章 結論以及未來工作	55
9.1 結論	55
9.2 未來工作	56
參考文獻	58

圖目錄

[圖 1] NPR 中的各種繪畫風格.....	5
[圖 2] 不同資料來源所生成的水彩繪畫	6
[圖 3] 各種水彩畫筆與其筆觸形狀	8
[圖 4] 各種水彩技法[10].....	9
[圖 5](a)色相環[11]，(b)(c)水彩的三原色與二次色[9].....	12
[圖 6]水彩繪畫中用高光處來表現物體受光面	14
[圖 7] Curtis[5]等人提出模擬水彩在紙上成色的三層流體模型	15
[圖 8] Jos Stam 流體模擬 Model[24]	16
[圖 9] LBM 流體模擬 Model[28].....	16
[圖 10] Lattice Boltzmann Method 晶格模型	18
[圖 11] Stream step.....	19
[圖 12] Collide step.....	19
[圖 13] System architecture 列舉幾個系統中重要元件的關係.....	22
[圖 14] 系統流程。包含前置處理部份以及單次影格處理程序。	23
[圖 15] Adapter 資料匯入示意圖.....	25
[圖 16] 判斷 X-file 中有 Bone 與否.....	26
[圖 17] 兩個 modified Z-buffer 的主要資料結構.....	27
[圖 18] 綜合三種線段產生出來的 Output	28
[圖 19] Adapt Depth Test 流程圖.....	29
[圖 20] Depth-Test 前後結果圖。	29
[圖 21] 執行完 Hidden Line Elimination 後的結果圖.....	30
[圖 22] 水彩模擬流程圖	32
[圖 23] stream step simulation 前後圖	34
[圖 24]不同沉澱比例所造成的筆觸樣式	34
[圖 25]一滴水在虛擬畫布上的擴散過程	36
[圖 26]一筆水彩的繪畫過程.....	36
[圖 27]RGB 色彩系統	37
[圖 28]CMYK 色彩系統	37
[圖 29]CMYK 三原色兩兩混色圖。	38
[圖 30]GUI 設計流程圖.....	39
[圖 31]GUI 介面.....	40
[圖 32]使用畫筆調整 Model 的方向。	41
[圖 33]使用提供的調色盤進行手繪。	41
[圖 34] Color Chart 系統流程圖	42
[圖 35]自動筆觸生成流程.....	43

[圖 36]筆觸的細節	44
[圖 37] LIC 示意圖。	44
[圖 38] 2D shape-based stroke vector 存取示意圖。	46
[圖 39] 柱狀體上的筆觸方向，為建模軟體中的基本型體之一。	46
[圖 40] 手掌的骨架階層構造	47
[圖 41]3D model-based stroke vector 存取示意圖.....	48
[圖 42]手掌上，Thumb 部分的三維筆觸分布.....	48
[圖 43] 具有 specular 效果的影像	49
[圖 44]水彩描繪過程示意圖。	49
[圖 45]手繪筆觸模擬的流程圖	50
[圖 46]手繪筆觸使用 LBM 在虛擬畫布上的模擬.....	51
[圖 47] 左圖為採用自動繪畫為基底，有明暗度的水彩畫。右圖為任意的水彩繪畫。	51
[圖 48] GUI 介面	52
[圖 49]使用自動繪畫的 GUI 介面	52
[圖 50]Teapot 描繪過程.....	53
[圖 51]線框架	54
[圖 52]在線框架直接繪圖的手繪水彩繪畫	54
[圖 53]以自動為基底的手繪水彩繪畫	54

第一章 序論

1.1 研究動機與目的

在電腦圖學的領域中，對於筆觸的模擬及研究由來已久，其中對此著墨甚多的，就屬 Non-Photorealistic Render(NPR)這一塊領域了。

而近年來，GUI 的設計與開發的重要性也逐漸受到重視，相較於缺乏 GUI 設計的軟體工具，能讓使用者專注在於利用軟體工具的輔助來完成工作。而隨著手寫板、觸控式螢幕等等硬體的開發及支援，GUI 與硬體的結合更提供給使用者一個直覺的控制及創作環境。尤其在於美工設計的領域來說，繪圖板的輔助相較於滑鼠與鍵盤更能讓使用者發揮所長。

在水彩模擬的領域當中，可由來源分成兩種模擬方式，其一為 2D image-based Method，其二為 3D model-based Method。由其對產生影像的處理，又可分為互動式筆觸模擬以及純粹的畫面水彩化。而這些水彩模擬的發展，都各有優缺點。注重在繪畫過程的筆觸模擬因為著重水彩在紙上的作用，而將重點專注在即時模擬水彩擴散筆觸的部份，雖然能讓使用者有更擬真的體驗，但也只如同傳統的繪畫方式，只在一張白紙或採用一個由 2D image 畫面水彩化所得的畫面作畫，雖然節省即時水彩生成過程的時間，對一個初學者而言，電腦無法發揮所能有的更多的輔助。而 3D model-based Method 以 CG 的方式提供給使用者關於更多場景知識的參考，例如光影的相對位置，物體的相互位置等等，相較於 2D image-based 較能夠提供給初學者更多的輔助。

因此，我們提出這篇論文，希望除了能提供給使用者一個能夠使用 CG 技術的來源場景外，也讓使用者能夠以更直覺的 GUI 即時水彩模擬繪圖作畫，不只讓專業使用者能夠發揮創意，也能讓初學者對水彩繪畫有更進一步的學習體驗。

1.2 論文編排模式

為了能夠生成「高可信度」的水彩模擬結果，我們先從其色彩、技法與畫法來分析，抽取出可供量化的參數，並探討其相關的色彩系統參考。此一部分將於稍後的第二章來做探討。

在第三章，我們將簡述近幾年計算機用來模擬流體的一個主要方法，Lattice Boltzmann Method，也是本篇論文用來模擬即時流體的方法。

第四章將介紹我們早期系統的一些基本架構，包括系統架構與資料讀取時所處理的前置工作的介紹，並講解系統自訂的虛擬畫布：Modified Z-buffer。此一 Z-buffer 亦可視為本系統在運作上的核心。

第五章講解水彩繪畫模擬的過程。包括了如何使用 LBM 模擬水彩擴散的流程解說與色彩轉換的部分。第六章則說明 GUI 的設計，繪圖板及 Color chart 系統的整合。第七章則說明筆觸模擬的作法，這裡主要分為兩個部分，即為使用自動筆觸模擬 2D shape-based Stroke 與 3D model-based Stroke，以及配合繪圖板的 Sketch-based 筆觸，包括其筆觸的生成，以及互動式的色彩混色等等。

最後，以第八章展示我們的研究成果，第九章為結論以及未來工作的說明。

1.3 本論文的貢獻

對於學術研究、國家發展及其他應用方面預期之貢獻。

學術部分：

- (A). 此研究以 3D 的場景來表現 Object 經場景所提供的資訊影響後，應在水彩畫中呈現的效果。一般的水彩畫作多是以平面資訊來作畫，鮮少以 3D 場景資訊來作畫，若本研究能夠有理想的結果，相信可以在數位藝術方面做出許多的貢獻與突破。
- (B). 根據水彩的特殊性質，我們可以建構出具有強烈水彩風格，而又不失 object 之特徵性與整體性的輸出影像。由於現有的 NPR 技術主要的應用仍侷限於 tone-rendering 的各種改進版本，即賽路路風格的卡通成像，或是偏向擬真的真實鋼體模擬成像，少見有完全以水彩筆觸構成的圖片。因此本研究若能有理想的成果，相信可以在技術部份作出許多的突破與貢獻。

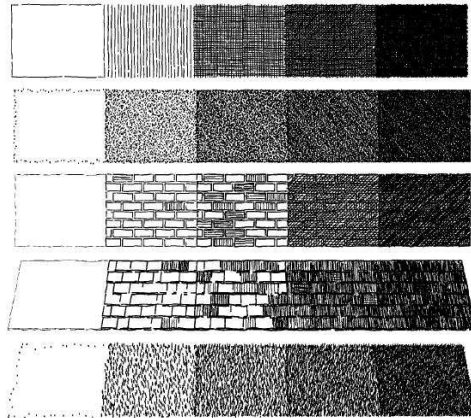
其他應用：靜態電腦繪圖

- (A). 本研究成果提供自動化的水彩筆觸影像生成，可提供水彩繪畫初學者一個學習繪畫筆觸的系統。
- (B). 藉由結合繪圖板以及 Color Chart，可以提出一個更直覺的水彩繪畫環境以及有依據的配色參考，用以輔助學習水彩繪畫技巧。
- (C). 藉由各種 plug-in 的開發，我們可以將此系統導入 3D-成像引擎中，給這些工具原先的使用者另一種完全不同以往的成像風格與更多的創作空間。

第二章 研究背景

2.1 NPR 概述

Non-Photorealistic Rendering(NPR)在圖學的領域中由來已久，此一特殊的成像方式不以 object 的擬真度為優先考量，而是著重於筆觸(stroke)、繪畫風格(paint style)等的呈現[1][2]，使產生的影像呈現出像是鉛筆[3]、油畫[4]或是水彩畫[5]等類似用手工繪畫出來的風格。其成像多以誇張的手法呈現欲突顯的部分，而不注重其呈現出的正確精確性。



(a) 素描鉛筆畫[3]



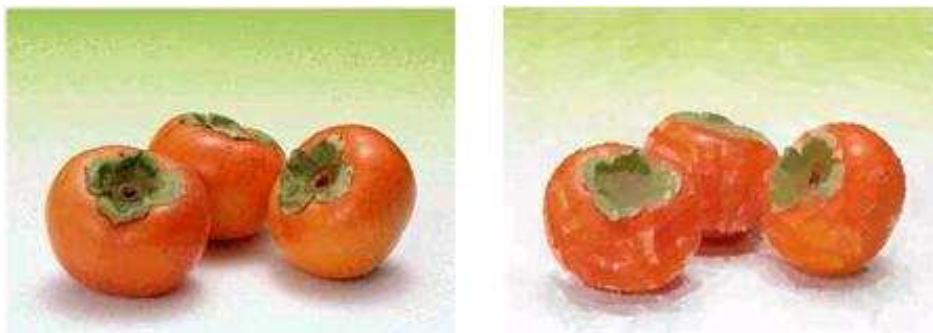
(b) 油畫[4]



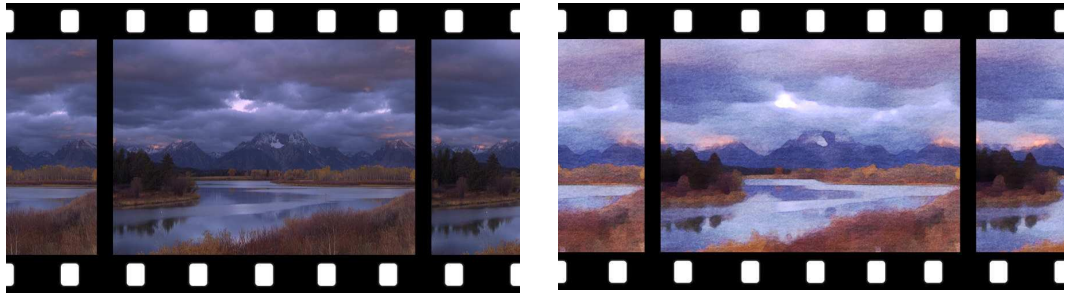
(c)水彩畫[5]

[圖 1] NPR 中的各種繪畫風格

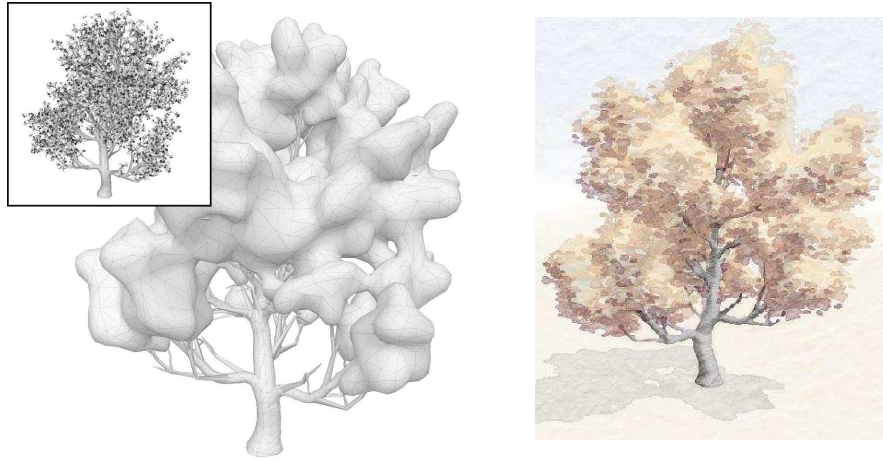
在 NPR 的領域中，根據資料來源劃分，大致上可以分為兩類：2D image-based method[2]以及 3D model-based method[6]。2D image-based 的 NPR 根據取得的資訊生成各種形式的筆觸，其原始資訊樣本為單張的 image[7]或是連續張數的 video frame [8]。所使用的處理手法其相關演算法亦較偏向於 image-processing 的領域。另一方面，3D model-based 的 NPR 則是以 3D-model 以及場景為資料來源[6]，於這些模型定義都是長久以來在 CG 領域中所廣為使用，因此有許多相關領域上的研究成果(如 3D-transformation, lighting, shading, etc.)都可以直接帶入以提供不同的參考及貢獻。相較 2D image-based 較難辨識背景及物件以及只能提供一個相同且不變的影像，3D model-based 能夠明確的分辨空間並擁有一個可供變化的場景，憑藉著不同的物件及場景條件搭配，給予使用者更多的彈性，提供給使用者一個不一樣的藝術創作空間。



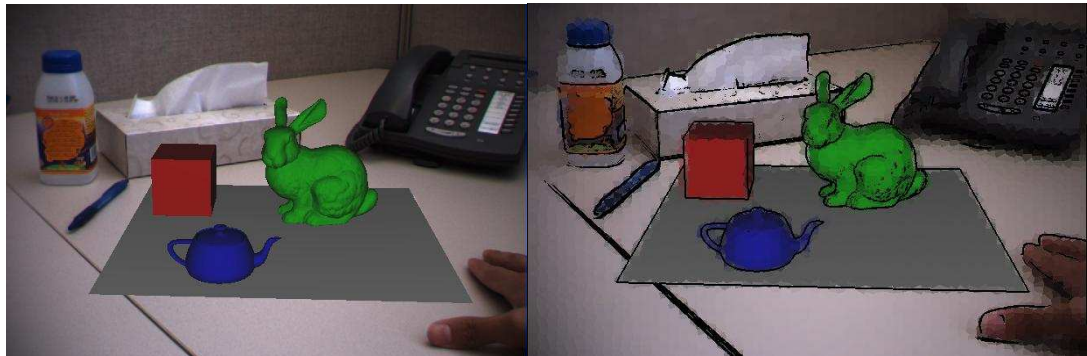
(a)單張 Image[7]



(b) Video Frame [8]



(c) 3D Model-Based [6]



(d) NPR for AR [20]

[圖 2] 不同資料來源所生成的水彩繪畫

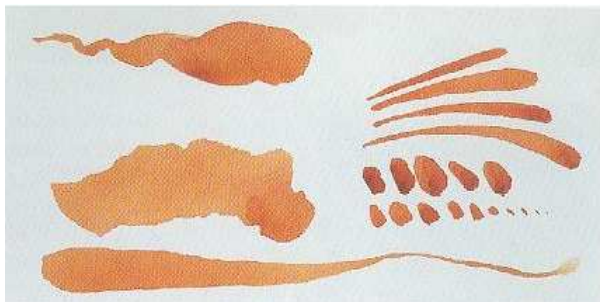
2.2 水彩概述

「水彩畫」簡單來說，就是使用水與顏料，以不同比例混合，在紙上作畫。因此，水彩畫的範疇可以推展到古埃及人的畫卷，波斯人富有異國情調的細密畫等，隨著時間的推演，歷經十八、十九世紀歐洲水彩畫的興起，尤其是英國多位水彩畫家們的努力，到十九世紀末，水彩畫已經發展出完整的獨立體系。

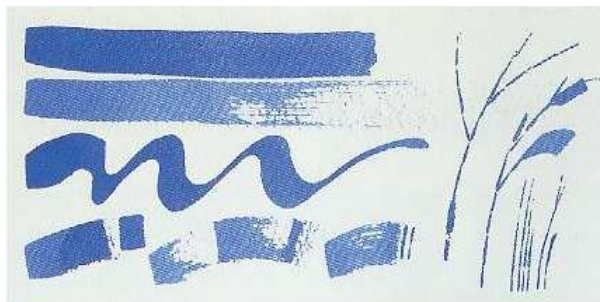
2.2.1 水彩工具與材料

創作一幅水彩畫，最基本的工具有畫筆、紙與水彩顏料。

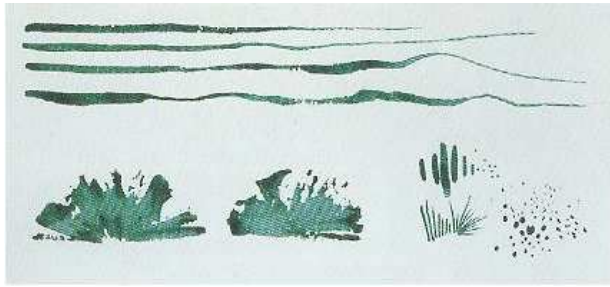
水彩筆一般可以分成圓筆、扁筆、線筆以及特殊用途的扇形筆等等[9]。圓筆是水彩畫的主要用筆，筆腹圓、筆頭圓而微尖，可以描繪任何物體。



(a)圓筆：使用在畫圓形以及必須薄塗顏料時。



(b)扁筆：可用來畫乾筆法的乾硬線條，能夠形成長度相同、重複排列的線條。



(c)線筆：在描繪細線時使用。

[圖 3] 各種水彩畫筆與其筆觸形狀

一張好的水彩畫紙可以讓畫者自由地揮灑，自在的地使用各種技法，創作出一幅滿意的水彩繪畫。水彩畫紙成分主要可分成碎布紙與木漿紙兩種。除了成分外，畫紙還有重量與開數的分別；以重量來說，畫家最常用的為重磅紙，開數則有全開、對開與四開等的區別。

水彩顏料是由顏料粉(成份依色彩而異)、阿拉伯膠(黏著劑)、甘油、蜂蜜或糖、牛膽汁(添加劑)，還有水(稀釋液)等構成，這些成分通常是由礦物以及動植物中提煉出來的。其中，顏料粉幾乎大同小異，差別只在於使用的溶劑、黏著劑和添加劑的不同。水彩顏料也有透明與不透明之分。而水彩畫常用的色彩約有三十八種。

2.2.2 水彩技法

在所有的繪畫技法之中，水彩畫技法變化可說是最大也最多樣化。探究其原因，「水」無法輕易控制走向的特性，是其主要的影響關鍵。

「平塗法」是最基本的水彩技法。簡單來說，就是將畫面分成幾個區域，每個區域平均地塗上一種顏色，不做明暗與筆觸轉折等等變化。其美感是完全靠色塊與色塊間的關係表達出來。

「縫合法」就是將兩個並置的色塊像縫衣般的接合在一起。

「乾筆法」又名乾擦法，其技巧就是以接近乾燥的畫筆沾上顏料，在粗糙的紙上

作畫，表現出紙與筆如擦拭般的筆觸。

「重疊法」以平塗法為基礎。藉由一次次在整張畫上同時畫上重疊的筆觸，由面而線而點，由淺而深地做疊色，繪出明暗、立體空間與筆觸相互交錯的美感。

「渲染法」又名溼中溼，必須先將紙打溼或者浸泡在水中，然後再上顏色，一直到最後，整張畫都是在潮濕狀態下完成的。渲染法完成的作品並不講求外型的精確，重要的是傳達一份浪漫情趣。渲染法由於都是朦朧的輪廓組成，所以如果能在幾乎完成的畫乾了之後，薄施小部分重疊筆，渲染法將更完美。

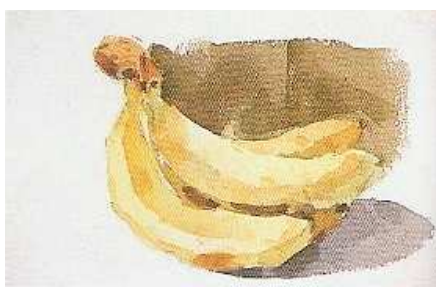
「複合技法」就是綜合運用上述的技法來完成作品，如此也能夠使作品更有變化。



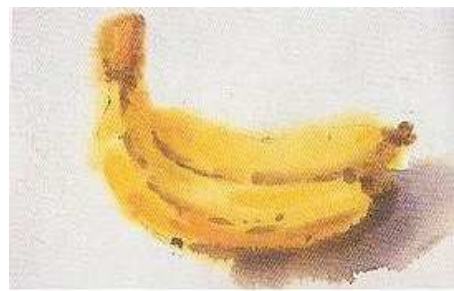
(a) 縫合法



(b) 乾筆法



(c) 重疊法



(d) 渲染法

[圖 4] 各種水彩技法[10]

2.2.3 色彩與配色

繪畫作品中的色彩是一幅彩色畫作不可或缺的元素。在同一幅畫作上使用不同的色彩分配，就能使這個畫作表現出不同的感覺。而畫面色彩常常講求所謂的"配色調合"，換句話說，就是講求將兩種以上的顏色做搭配，以產生出新的視覺效果，但是當這些顏色放在一起時，彼此是要能夠互相搭配並產生共鳴，而不會太過突兀而造成視覺上的衝突。但是也有許多藝術家的用色搭配追求的是大膽創新，這樣的色彩搭配，有時雖然讓人匪夷所思，但也創造了不同的配色思維。

這種對顏色的感覺，牽涉到人們眼睛觀看共同存在的色彩時，不會僅對各單一的色彩發生感覺，而是同時對所有相關色彩，共同產生「同時對比」的色彩感覺。而觀者眼中的這種「同時對比」色彩感覺，又分成「色相對比」、「明度對比」、「彩度對比」三種「色彩對比」感覺。所謂「色相對比」、「明度對比」、「彩度對比」色彩感覺，是指幾個色彩同時出現的「色相相互間關係之對比」、「彩度相互間關係之對比」、「明度相互間關係之對比」的影響。

所謂的色相，是指光線照射在物體上之後，依據不同反射光的波長在光譜上的位置所代表的名稱；簡單來說，就是色彩的通用名稱。譬如，"藍色"就是一種色相。而所謂的「色相對比」，就是色相間的差別產生的對比。

明度就是色彩的明暗程度(亮度)，所以不同的顏色就有不同的亮度。譬如天藍色較寶藍色亮度高、鵝黃色比土黃色亮度高。在無色彩中，白色為最高明度，黑色為最低明度，在白色與黑色之間，還有各種不同的灰色，構成所謂的「明度系列」。而有色彩的明度變化，就以光譜來說，黃色明度最高，而藍紫色明度最低。當一個顏色混入的白色越多，明度越高，當混入的黑色越多，則明度越低。把兩個不同明度的顏色排列在一起時，能使較明亮的顏色變的更亮，較暗淡的顏色變的更暗，這種現象即可

稱為「明度對比」。如把同一明度的灰色置於白底與黑底上，會感覺到白底上的灰色較暗，而黑底上的灰色較亮。另一種常見的明度對比影響，稱為「邊緣對比」。就是兩色相接時，在相鄰的邊緣處會產生特別強烈的對比現象。例如將同一亮度的灰色放在黑白兩色當中比較，和白色相接的灰色部分看起來較暗，而與黑色相接的灰色部分看起來較明亮。在許多整齊的暗色小方格上，隔以明色線條時，會在白色線條交叉處感覺到暗色陰影的存在，這也是邊緣對比的現象。明度對比比其他任何對比的感覺都要強烈，所以影響畫面的效果也最大。在一個畫面中如何安排不同明度的色塊也可以幫助表達畫作的感情，如果天空的明度比地面低，就會產生壓抑的感覺。但是極端的明暗對比，會產生不安與煩躁的感覺，因為眼睛的瞳孔看到暗色時會擴張，看到亮色時會縮小，眼睛不斷的適應亮與暗的強烈對比時，會因調節過度而產生煩躁與不安感。

彩度(亦稱飽和度、色度、純度)，指的是色彩飽和的程度或色彩的純粹度。純色因為不含任何雜色，飽和度及純粹度最高，因此任何顏色的純色均為該顏色中彩度最高的顏色。純色加上其他的顏色，都會讓其飽和度下降，不論黑、白、灰或彩色，其彩度都會降低。混合越多，彩度越低。例如純色加上白色，明度會提高，但是純色及白色的飽和度都下降；純色加上黑色，純色的明度下降，而且純色及黑色的飽和度都降低。而所謂的「彩度對比」，即是將彩度高、低不同的顏色並置，所產生的對比現象。如將彩度相同的綠色，放在彩度較高及較低的綠色上，則放在彩度較高的綠色，會因對比的關係而顯得彩度降低，反之放在彩度較低的綠色，會顯得彩度較高。

在水彩的色彩系統中，紅、黃、藍這三種顏色是無法由調色取得的，此三種顏色稱之為「色彩三原色」，在繪畫時，可以利用他們調配出所有的顏色。再加之以上所述的各項色彩配色基礎與水彩技法，來創作水彩繪畫。



(a) 伊登十二色相環:中間三角形即三原色，每一邊上為二次色，最外環為三次色



(b)三原色：紅、黃、藍



(c)以二個原色調出二次色(濕重疊)：黃和紅調出橙色，黃和藍調出綠色，而紅和藍則調出紫色

[圖 5](a)色相環[11]，(b)(c)水彩的三原色與二次色[9]

2.3 水彩畫法的分析與量化

要完成一幅水彩作品，必須先構圖，之後再加上色彩。在構圖方面，沿用了之前的研究結果，在研究中將 3D-model 投影在 2D 的虛擬畫布上，所以我們可以在虛擬畫布上得到整個 3D 場景中物體的輪廓線。所以隨後的加入色彩部分即是我們本節所要分析的重點。

水彩畫的技法，豐富且富有變化，畫家會依當時畫面所需而選擇技法，所以一幅畫常常是採用複合技法來完成。而每一幅畫的創作，也因為畫家不同的感覺，而有不一樣的變化。這些變數，都增加了我們對水彩分析與量化的困難度。

大體而言，每種物體其質感各不相同，如粗糙、笨重、細膩、光滑等，善用描繪方式的不同，可以利用明暗的變化及筆觸的風格來達到這些題材的特色。以下大略的分出兩大類水彩筆觸。

(A) 以模型形體為主的筆觸。在這裡，筆觸所扮演的角色，往往是用來強調物件的三維特性，讓觀者能由 2D 的繪畫在腦海中重建出物件原始的 3D 型態。在本研究中我們用物體本身的 Skeleton 特性來計算此一筆觸。Skeleton 是 3D-model 一項很重要的形狀特徵，此資訊容許我們根據 Skeleton 重建 3D-model 的形體，也常應用在許多不同的領域。

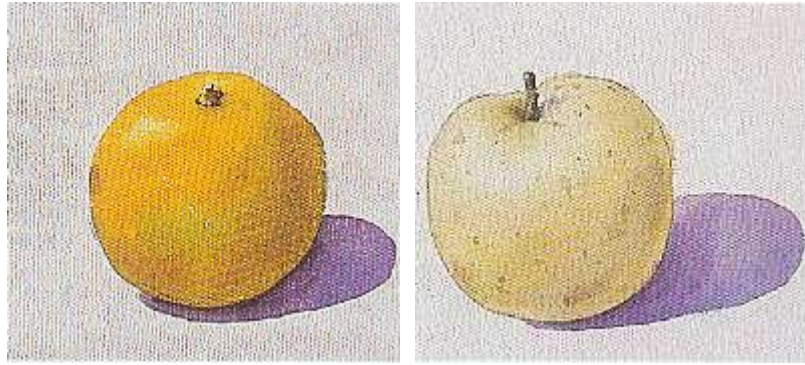
(B) 根據 3Dmodel 投影後的 2D-image Shape 所產生的筆觸。在繪圖的時候，有時我們需要額外的線條來強調物體本身 2D 的形狀。這種繪畫手法尤其使用在當物體本身具有不規律起伏的表面時。大致來說，此種繪線常會隨著物體本身輪廓線起伏生成。

另一類為：

(A) 參考物體的特徵所產生的筆觸。一個物體的特徵往往是用來辨別物體種類的方法，例如：skeleton、edge 或 color 都屬於物體特徵。

(B) 依照物體 texture 所產生的筆觸。物體的 texture 也就是物體的顏色與紋理。

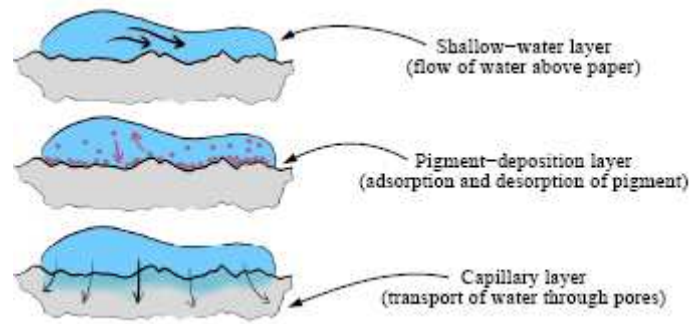
除了筆觸的分析外，明暗度也是水彩繪畫上一個重要的環節。在繪圖時常需要考慮到物體上的明暗層次，以便做出物體的立體感。所以常會在物體最亮的部份留白或是以清水洗潤亮面的高光處。



[圖 6]水彩繪畫中用高光處來表現物體受光面

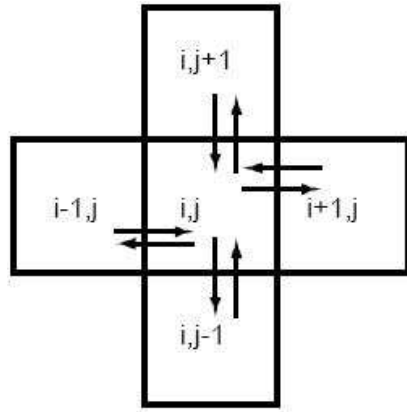
2.4 相關研究

水彩模擬由 *Small*[12]發展至今已將近 20 年的歷史，雖然 *Small* 的研究並未被廣泛使用，但 *Curtis*[5]運用並改進其想法，利用電腦模擬物理模型產生出水彩在紙上的上色過程：首先模擬了水在紙上流動的狀況，其次模擬顏料在水流過區域的沉澱情形，最後模擬其毛細現象所造成的結果。之後也有許多的水彩模擬系統，包含讓使用者自行繪圖 [13][14]，或是以影像處理的方式模擬 NPR [7][8]，或是以 3D model-base 的方式做 NPR 水彩模擬[6][15][16][17][18][19]，甚至應用在即時的 AR 系統上[20]。在這裡值得注意的是，*Curtis* 使用的流體模擬是採用 Navier-Stokes equations[21][22]，而大多數的水彩模擬考慮到執行速度的問題，大多採用 texture 的方式模擬流體造成的現象[6][8][16][19][20][23]，或是 Wyvill Noise[17]，或是即時座標[15]。相較於 *Curtis* 使用的物理方法，雖然輸出畫面的時間縮短許多，但是對於即時的流體模擬現象，卻受到了相當的限制，譬如使用 texture 模擬[16]，是將影像做平滑化(Smoothing)之後，再使用 texture 模擬水彩筆觸，這樣的方式無法使用在模擬單一水滴在紙面上渲染的效果。

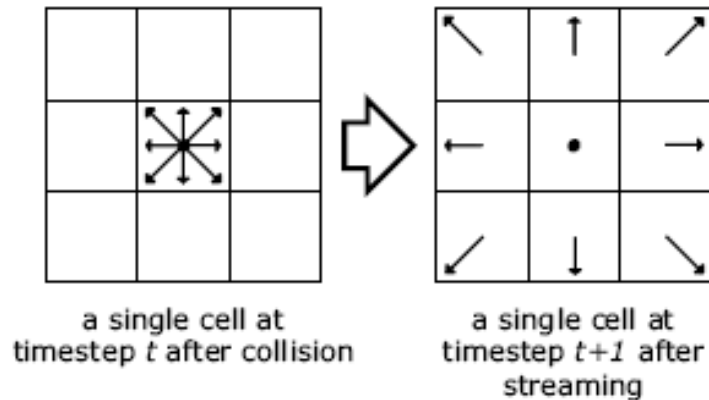


[圖 7] Curtis[5]等人提出模擬水彩在紙上成色的三層流體模型：第一層模擬水在紙上的流動，第二層模擬顏料的沉澱，第三層模擬毛細現象

而相對於使用 texture 輸出水彩模擬畫面的方法，專注在繪畫過程的 2D 水彩繪畫模擬系統[13][14]則選擇使用物理模型來做水彩模擬。Van Laerhoven[13]延伸 Curtis 的研究，但是採用 Jos Stam 的流體模擬[24]，Jos Stam 使用符合電腦思維 cell 模式來模擬流體，避免了繁複的 Navier-Stokes equations 計算，但是 Jos Stam 只提供 5 個 cell 的計算，忽略了斜邊方向的 cell。Chu and Tai[11]延伸其原本的水墨畫系統[25]，加入 Curtis 研究中所使用的 Kubelka-Munk (KM) model[26][27]，將水墨畫模擬轉換成水彩模擬。Chu and Tai 所發展的水彩模擬雖然還未完善，但其延伸使用水墨畫系統[25]中所使用的 lattice Boltzmann equation(LBM) [28]，得到了相當有效果的模擬。LBM 類似於 Jos Stam 所發展的流體模擬，但其使用了 9 個 cell 計算平面流體，並增加了較準確的碰撞、黏度等等條件的計算。



[圖 8] Jos Stam 流體模擬 Model[24]



[圖 9] LBM 流體模擬 Model[28]

在利用 Lattice Boltzmann Method 來實做繪畫方面，有香港科技大學 Nelson Chu 博士所發表用來模擬中國水墨畫風的 MoXi 系統[29]。他們模擬墨水與吸收性良好的紙之間的相互作用，他們將 paper-model 分成三層，分別為 surface、flow 與 fixture layer，主要是利用 Lattice Boltzmann Method 模擬墨水在 flow layer 的流動與擴散現象。在成像方面，主要是建立了一個跟使用者互動的介面，使用者使用他們所提供的介面再利用手寫板便能在螢幕上呈現出自行繪畫且具有水墨畫風的影像。而近年來，他們擴展了顏色系統，參考 Curtis 研究中所使用的 Kubelka-Munk (KM) model[26][27]，發展了一套水彩繪畫系統[14]。

但是，對於結合 3D model-based NPR 所能提供的豐富場景變化及 2D 繪畫系統能展現的即時水彩著色過程模擬等等相關研究卻在少數。*Bousseau*[23]的研究提供使用者 2D image-based 及 3D model-based 資料輸入，並讓使用者能夠更改物件顏色及繪畫技巧，但系統是使用 texture 模擬顏料粒子的分布，對於模擬流體產生的渲染效果有所限制，也無法以動畫的方式展現即時的水彩著色模擬。

以上所探討的研究，與我們研究最大的差別，除了模擬的對象不同之外，還有成像方面，我們的系統主要是能夠讀入 3D-model，然後根據 model 上的資訊，便能夠自動化的產生出具有水彩筆觸風格的影像，而讀入 3D-model 的優點，我們的系統可以動態變化場景資訊，並結合動態手繪，生成出更具個人風格的水彩繪畫。

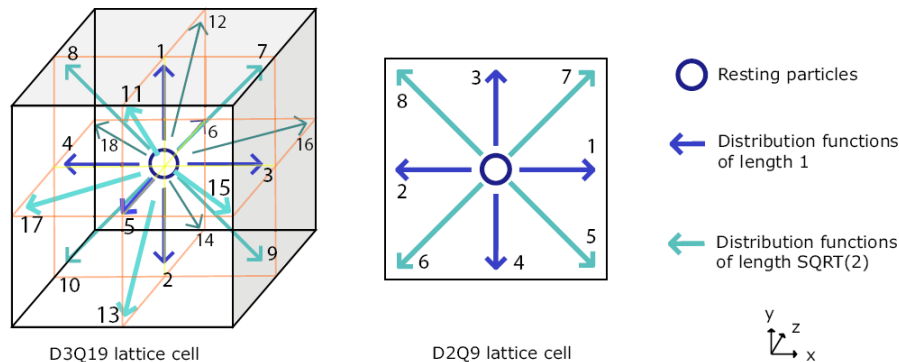
第三章 Lattice Boltzmann Method

本篇論文利用 Lattice Boltzmann Method 來模擬流體。Lattice Boltzmann Method 的優點[28]是不需要像 Navier-Stokes equations 處理複雜的偏微分方程式，而且只需要做簡單且局部的運作便能模擬流體的運動。

3.1 簡介

Lattice Boltzmann Method(LBM) 物理模型，是近幾年計算機用來模擬流體的一個主要方法。顧名思義它是以晶格為基礎而運作的模型。也就是說 LBM 是利用精簡的粒子運動模型來模擬流體。LBM 也可以被描述成是細胞自動機的一種類型，所有的細胞在每一個時間 t 中會依簡單的規則更新，而且每一個細胞的狀態都是由在上一個時間中($t-1$)包圍它的細胞的狀態來決定的。

LBM 模型是模擬不可壓縮流體，允許粒子只沿著晶格向量移動，常見的標準晶格模型為 D3Q19 與 D2Q9。在本研究中我們使用模擬水在 2D 平面上流動的 D2Q9 晶格模型，其中的 2 用來表示維度，而 9 則是用來表示有 9 個不同的晶格速度向量。如 [圖 10] 左圖為 D3Q19 的晶格模型，右圖為 D2Q9 的晶格模型

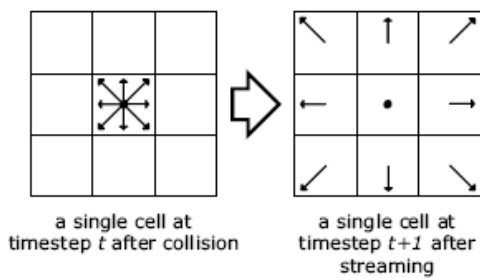


[圖 10] Lattice Boltzmann Method 晶格模型

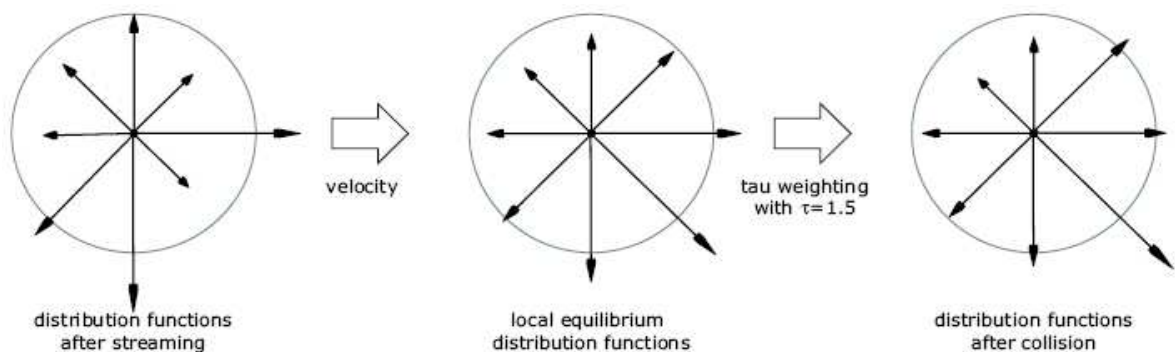
3.2 LBE 的兩個運作步驟

由[30]可知，在 LBE 模擬的過程當中包含了兩個步驟，而且在每一個時間 t 中都會重複這兩個步驟。第一個步驟為 stream step，第二個步驟為 collide step。stream step 描述晶格上粒子的移動；collide step 表示粒子在移動間所發生的碰撞。

我們由[圖 11]得知，stream step 只是將每一個晶格中，所有的 distribution functions 複製到相鄰晶格的相對向量上，其中的向量 0 是不會移動的，而在每一次的 stream step 當中，所有的晶格必須紀錄沿著每一個向量所移動的粒子個數，稱 particle distribution functions，通常用 f_i 來表示。在真實的流體中，粒子是不停的互相碰撞且互相影響的，所以 collide step 用來改變所有 particle distribution functions 的粒子分配，讓整個流體達到碰撞後的平衡。



[圖 11]stream step。左圖說明了在 D2Q9 中單一晶格分別在 streaming 前後的粒子分配。



[圖 12]collide step。在 collide step 期間，利用 stream step 所產生的 distribution functions(左圖)來計算 velocity 與 density，再利用 velocity 與 density 計算得到 local equilibrium distribution functions(中圖)。之後，兩個 functions 經由權重計算相加，便可以得到在 collide step 之後所以產生新的 distribution functions(右圖)。

在整個模擬過程中的兩個步驟，stream、collide step 在數學上可由以下兩個式子表示。 f_i' 表示為新的粒子分配函數， $f_i^{(0)}$ 為 collide 的平衡分配函數。

$$f_i' = (1 - \omega)f_i + \omega f_i^{(0)} \quad (1)$$

(1) 中的 ω 為一個 relaxation parameter，這個參數會影響流體的黏性， ω 越小越黏稠、越大越接近水的性質。

$$f_i^{(0)} = w_i \left\{ \rho + \rho_0 \left[3(\vec{e}_i \cdot \vec{u}) + \frac{9}{2}(\vec{e}_i \cdot \vec{u})^2 - \frac{3}{2}(\vec{u} \cdot \vec{u}) \right] \right\} \quad (2)$$

(2) 中的 w_i 表示每一個向量的 weight，當 $i=0$ ， w_i 為 4/9； $i=1,2,3,4$ ， w_i 為 1/9；

$i=5,6,7,8$ ， w_i 為 1/36。 \vec{e}_i 表示 9 個不同方向的晶格向量。 $\rho_0=1$ 。 ρ 表示為每一個晶格的密度(density)(3)。 \vec{u} 為速度(velocity) (4)。

$$\rho = \sum_{i=0}^8 f_i \quad (3)$$

$$\vec{u} = \frac{1}{\rho_0} \sum_{i=0}^8 \vec{e}_i f_i \quad \text{for } \rho_0 = 1 \quad (4)$$

第四章 系統概述

此研究，主要建構在我們早期所開發的 NPR System 之上[11]，所以本章將介紹早期 NPR System 的系統架構、資料讀取與虛擬畫布的開發。

4.1 系統架構及運作流程

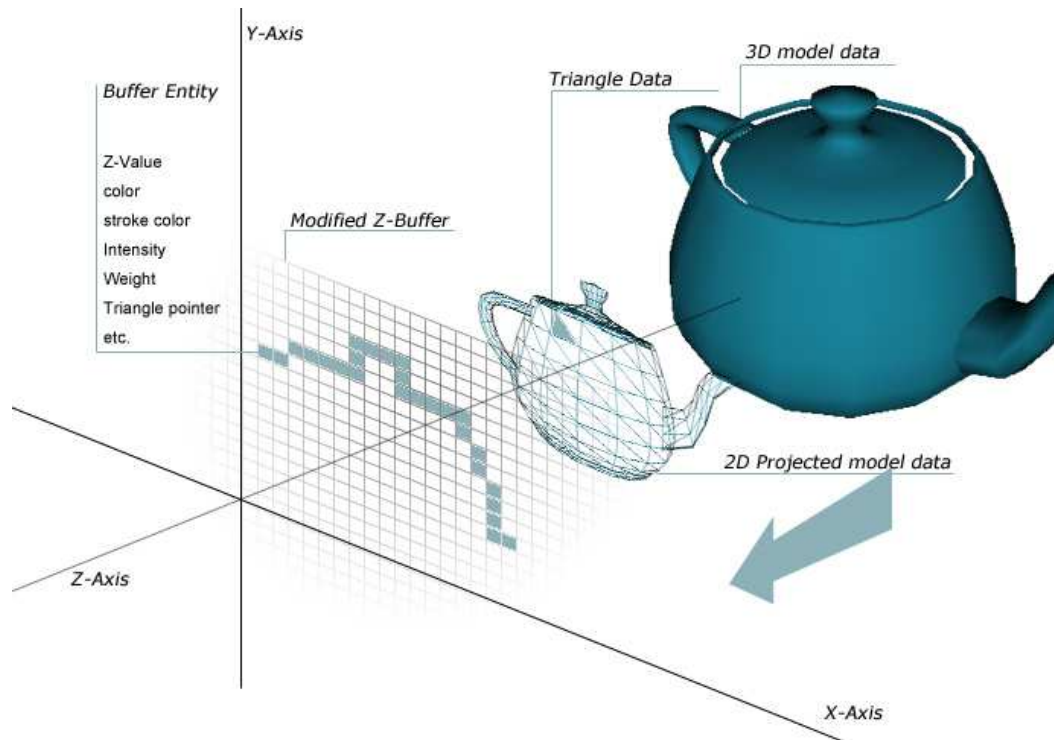
本節我們說明系統的構想、系統架構以及運作流程圖。

我們需要一個能夠模擬畫布的資料結構。這張模擬畫布的每一個 pixel，必須可以自由的參考 3D-model 以及矩陣運算過後的 2D-shape，或者整個世界場景中的相關資料如光影、方位等。基於這個構想，我們嘗試改良原本繪圖卡所提供的 Z-buffer 這樣的繪圖機制。藉由重新定義 Z-buffer 上的儲存資料結構來達到我們需要的結果。

4.1.1 系統架構

為了可以處理畫布上每一條筆觸的細節，我們建構出一塊由 Buffer 組成的虛擬畫布。此一畫布除了傳統的 pixel - Z information 之外，還必須能夠參考到原始模型的資訊，以及平面化後的 shape 資訊，以便於計算出合適的筆觸資料。

如[圖 13]所示，此為 NPR-System 的基本架構。3D model data 即為原始的模型資料；Modified Z-Buffer 即代表「虛擬畫布」。每一個 Buffer 上的像素必須儲存一個 Triangle data 的連結，以方便讓 Modified Z-Buffer 上的每一塊像素可以便利的參考 3D 原始資料，以及套用經過所有的轉置矩陣(模型座標矩陣、世界座標矩陣、投影矩陣)計算後的 2D projected model data。其次，為了整理出模型 Triangle-Triangle 之間的相互關係，以及便利找尋模型上的輪廓線，我們必須重新定義原始模型中三角形的 Index 資料，重新儲存為 Triangle data。



[圖 13] System architecture 列舉幾個系統中重要元件的關係

4.1.2 運作流程

概括來看，可以分為三個主要的部分，如[圖 14]。

(A) 3D model loader – Pre Processing part

我們將由 X-file 讀入的資料，重新拆解並整理成我們系統需要的資料結構。模型與骨架的階層關係、模型內部的表面關聯性都在此作前置處理。

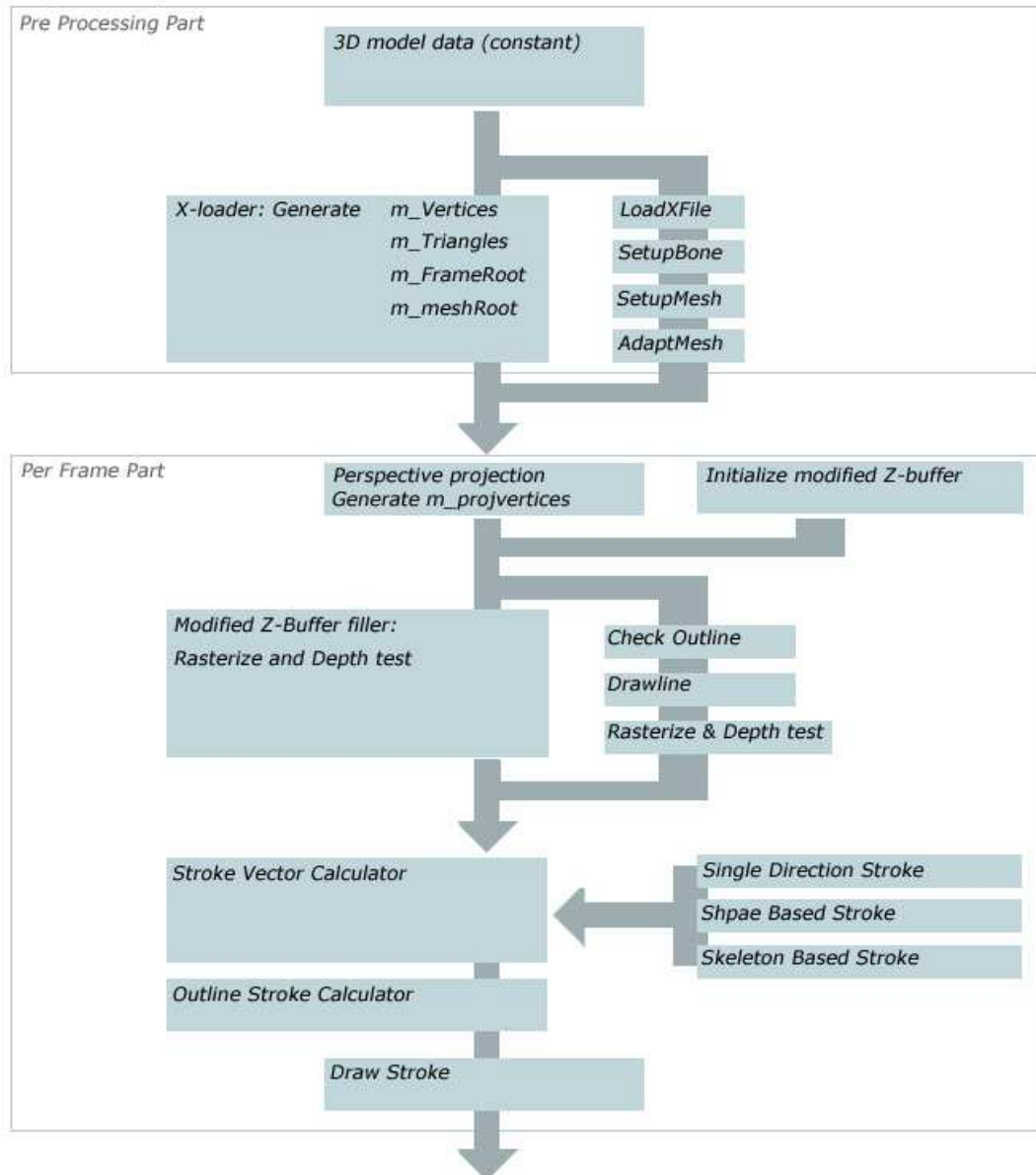
(B) Modified Z-Buffer: Rasterizing & Depth test – Per Frame part

1. 判別線段資訊。填入需要繪製的線段
2. 初始化 modified Z-buffer，並將 3D-model 的浮點資訊光柵化並填入 Z-buffer。
3. 深度測試，消除隱藏線。

(C) Stroke Calculator & Stroke Drawer

1. 選擇不同的筆觸生成方式，生成筆觸的走向。
2. 給定外框線段的權重。
3. 繪出筆觸效果。

Single frame procedure



[圖 14] 系統流程。包含前置處理部份以及單次影格處理程序。

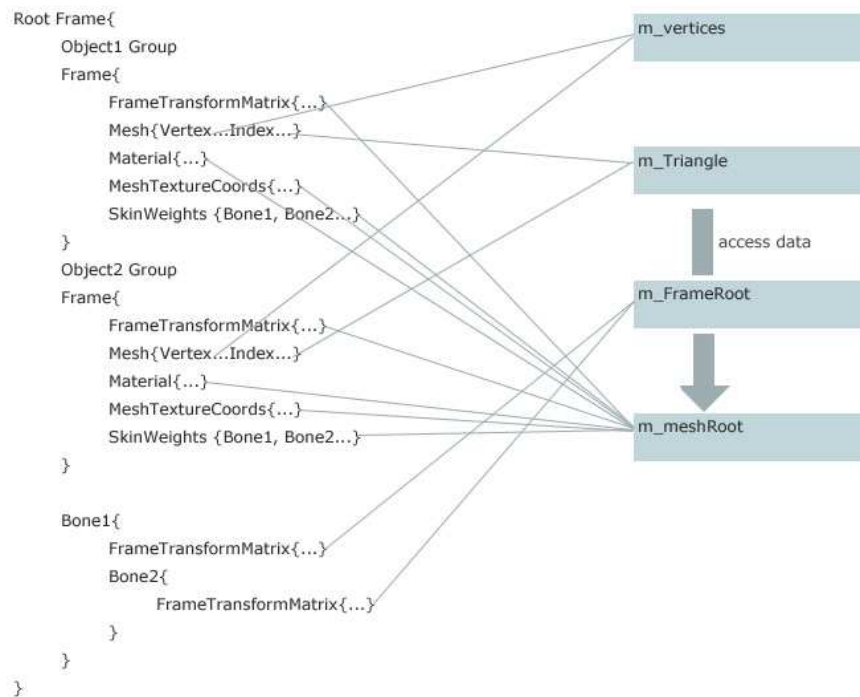
4.2 資料讀取及虛擬畫布

在本系統中，我們選擇使用 Direct3D 的 X-file 作為資料的儲存格式。另外，由於著名建模軟體如 Maya、3DsMax 皆有支援 X-file 的檔案格式，更便利於日後的相關開發。在虛擬畫布方面，我們使用自定的格式來實作一個 Z-buffer，即先前所提的虛擬畫布。使所有的 Stroke 計算以及生成均可以利用此畫布達到需求的效果。

4.2.1 資料讀取及整理

我們讀取進來的資料格式為 X-file 的資料格式，其優點為具有階層特性的物件關係。由於 X-file 中的頂點資訊為“non-unique”的特性，若沒有經過整理與剔除多餘的頂點的步驟而直接使用，將會大量消耗不必要的系統運算資源。

因此，為了可以讓資料更適於我們的系統使用，所以在前置處裡的部分亦增加一個 Data Adapter，用來將 X-file 的檔案資料轉換成我們的系統所需要使用的資料結構。在系統裡，我們將場景中的所有 vertex 全部匯入整理成一份 Vertex List，而將群組關係及 Triangle 特性交由 Triangle List 管理。包含 Index 資訊，群組資訊，骨架資訊等。因此必須將 Input 的 X-file 重新拆解、整理。



[圖 15] Adapter 資料匯入示意圖

(A) Vertices Data :

由各 Object 之 Vertex 資料取得。由於 Maya / 3DsMax 轉換的 X-file 其頂點資訊皆不具有唯一性，也就是說一個簡單的正四面體，卻會出現 $3*4=12$ 個頂點資訊。因此，必須先做過一次資料篩選以及建立 Replace Table 更改其 Index 參照資訊之後，才能將具有唯一性的頂點資訊填入 Vertices List。

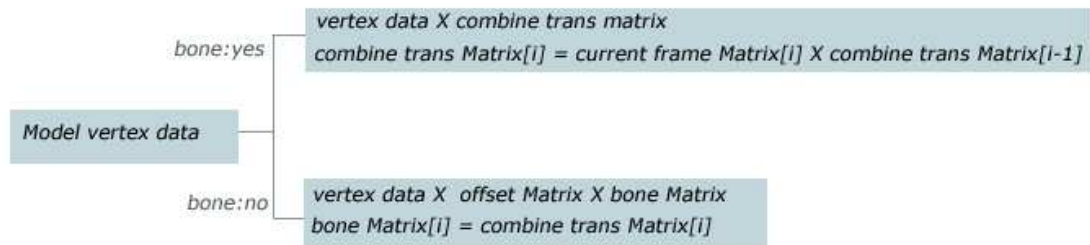
(B) Triangle Data :

根據上面的 Replace Table 以及物件間的 Offset Table 來更新資料資後，輸入到 Index List。

(C) FrameRoot :

儲存 Bone 的階層關係以及 Bone 的轉置矩陣、頭尾資訊等。

此外，當讀取的 X-file 僅有基本的模型資料時，此 X-file Adapter 也負責轉換此種類型的模型資料，模型中 Bone 的有無，將決定 Vertex 所需參照的 Matrix。



[圖 16] 判斷 X-file 中有 Bone 與否

4.2.2 虛擬畫布：Modified Z-buffer

我們使用自定的格式來實作一個 Z-buffer，即先前所提的虛擬畫布。使所有的 stroke 計算以及生成均可以利用此畫布達到需求的效果。

如[圖 17]所示，右側的表為 Buffer Entity，代表每一個螢幕上出現的像素。其中除了最基本的 Z value，也包含了關於此 Pixel 的權重(weight)、經過 3D 場景運算後的 Color、以及各種筆觸的走向(vector[i])及相對應的筆觸顏色(strokeColor[i])等。其中的 tPtr 則是用來連結所屬的三角形。透過此一指標，我們可以輕易的取得物體原本在 3D-model 所保留的資訊。

```

class mZbuff {
public:
    mZentry buf[HEIGHT][WIDTH];

    int getSize();
    void clear();

    int pointNum;

    mZbuff();
    ~mZbuff();
};

```

```

class mZentry {
public:
    D3DXVECTOR3    vector;
    D3DXVECTOR3    strokeVector[3];

    DWORD         color;
    DWORD         strokeColor[3];

    float         intensity;

    TRIANGLE      *tPtr;
    LPSTR         meshName;

    bool          exist;
    int           type;
    float         Z;
    float         weight;

    mZentry();
    ~mZentry();
};

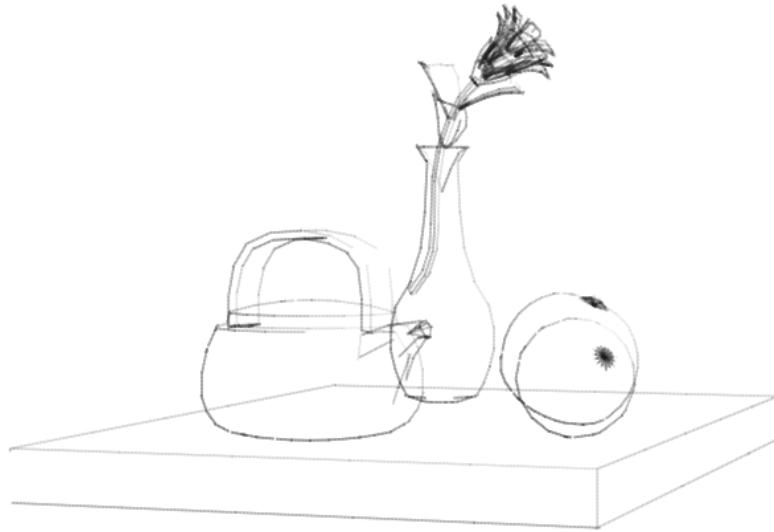
```

[圖 17] 兩個 modified Z-buffer 的主要資料結構

由於我們的參考物件為實體 3D 模型，因此整個物件的外框線(Outline)將根據生成方式區分為三種：Silhouettes, Boundaries, Discontinuities。

- (A) Silhouettes：當線段的相鄰兩三角形其中一個三角形法向量與視角夾角小於 90 度，另外一個大於 90 度，即代表這條線處於一個可視的三角形與一個不可視 (culled)的三角形中間。此種線段稱為 Silhouettes。
- (B) Boundaries：當線段只有一個相鄰三角形時，即代表這條線在模型的邊緣，此種線段稱為 Boundaries。
- (C) Discontinuities：當線段的相鄰兩三角形法向量夾角大於某個固定角度時，即判斷此線段需要繪出，稱為 Discontinuities。

然而，藉由上述三項線段結合所產生的輪廓線，會發生從被遮蔽的物體後方透出來的情形[圖 18]，因此需要將這些「應該被前物遮蔽的線條」從線段中剔除，稱為 Hidden-Line Elimination。

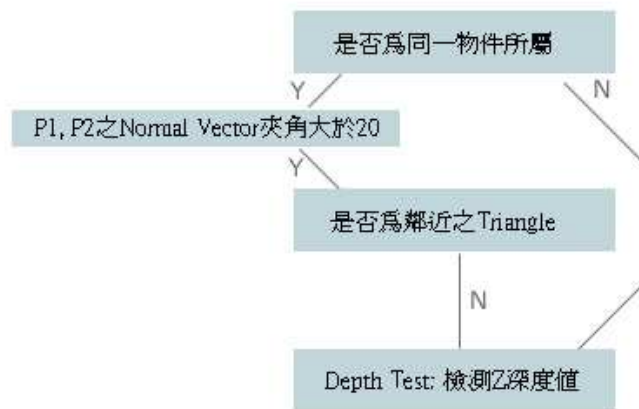


[圖 18] 綜合三種線段產生出來的 Output

為了消除這些線段，我們採用較為普遍的 Z-Depth Test 實作。對於場景中所有物件的 Triangle，利用 Rasterization，我們依序將其填入虛擬畫布(Modified Z-buffer)中，同時在每一個 Pixel 填入時，藉由比對 Z-buffer 中每一個填色物件的深度(Z value)，來決定最後此 Buffer 將歸於哪一個物件。

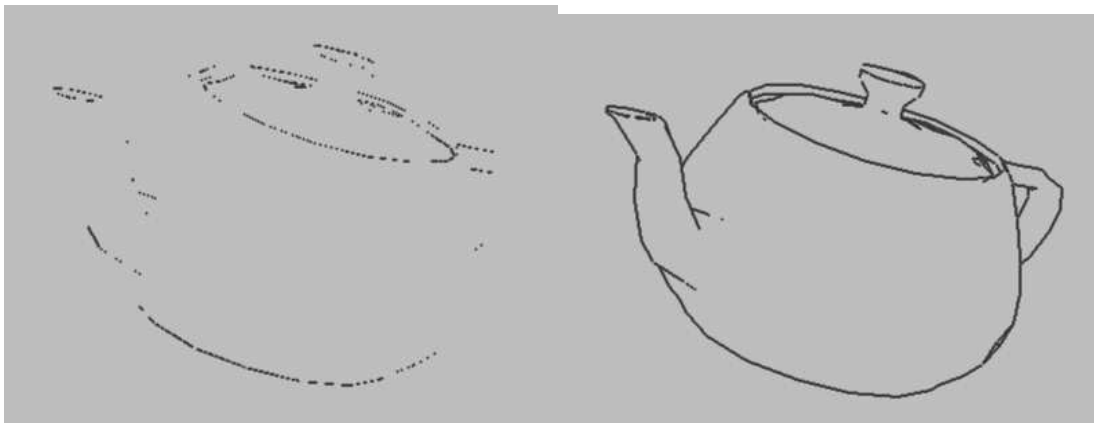
然而此一方法在實作時，對於我們的系統卻出現一個問題。我們將 modified Z-Buffer 單元分為三類：Empty、Texel、Edge。Empty 代表沒有被任何物件填色的 Pixel；Texel 代表內圖色塊；Edge 則代表物件的 Outline。傳統 Z-Buffer 在使用時，考慮到的乃單純物件 pixel 的深度，並不需要考慮內圖色塊(Texel)與邊緣線條(Edge)的分別。因此，當物件為“高複雜度、圓弧狀表面”時，便會出現問題。

當物件的平滑表面變成輪廓線時，由於過於平滑，用來判斷是否為 Outline 的 3D-model 邊緣會被自己的內圖色塊遮蔽(Z-Value 判斷後，texel 會遮蔽 Outline)。導致部份的 Outline 消失。為了解決這個問題。我們改進了 Depth-Test 的檢測方法。基本檢測流程如下：

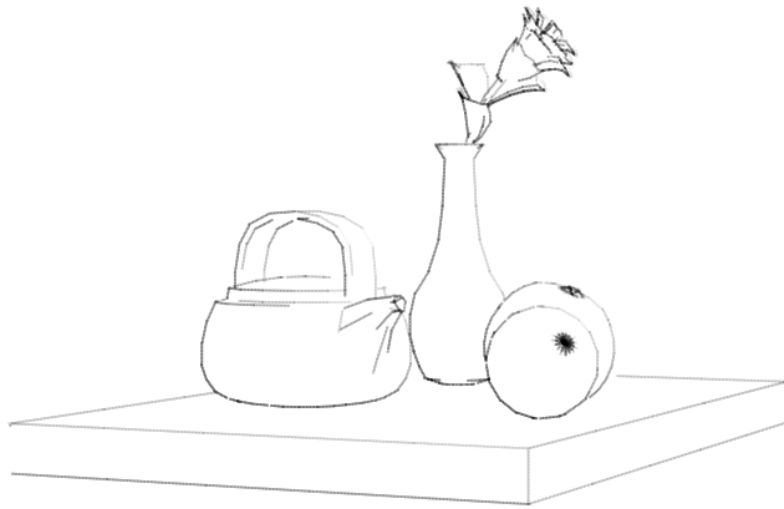


[圖 19] Adapt Depth Test 流程圖

- (A) 若 P1, P2 分別屬於不同的物件，則直接作深度測試。
- (B) 反之，則判斷 P1, P2 點上的 NormalVector(由 3Dmodel 取得)之夾角，若超過一預設之門檻(20 度)則繼續判定條件(C)
- (C) P1, P2 若彼此為非相鄰之 Triangle，則作深度測試。



[圖 20] Depth-Test 前後結果圖。左圖為套用傳統的 Z-Depth 的結果，Outline 被刪除的情形相當嚴重。右圖為改進 Depth-Test 後的結果，Outline 正常顯示。



[圖 21] 執行完 Hidden Line Elimination 後的結果圖

第五章 水彩模擬

5.1 Simulation Overview

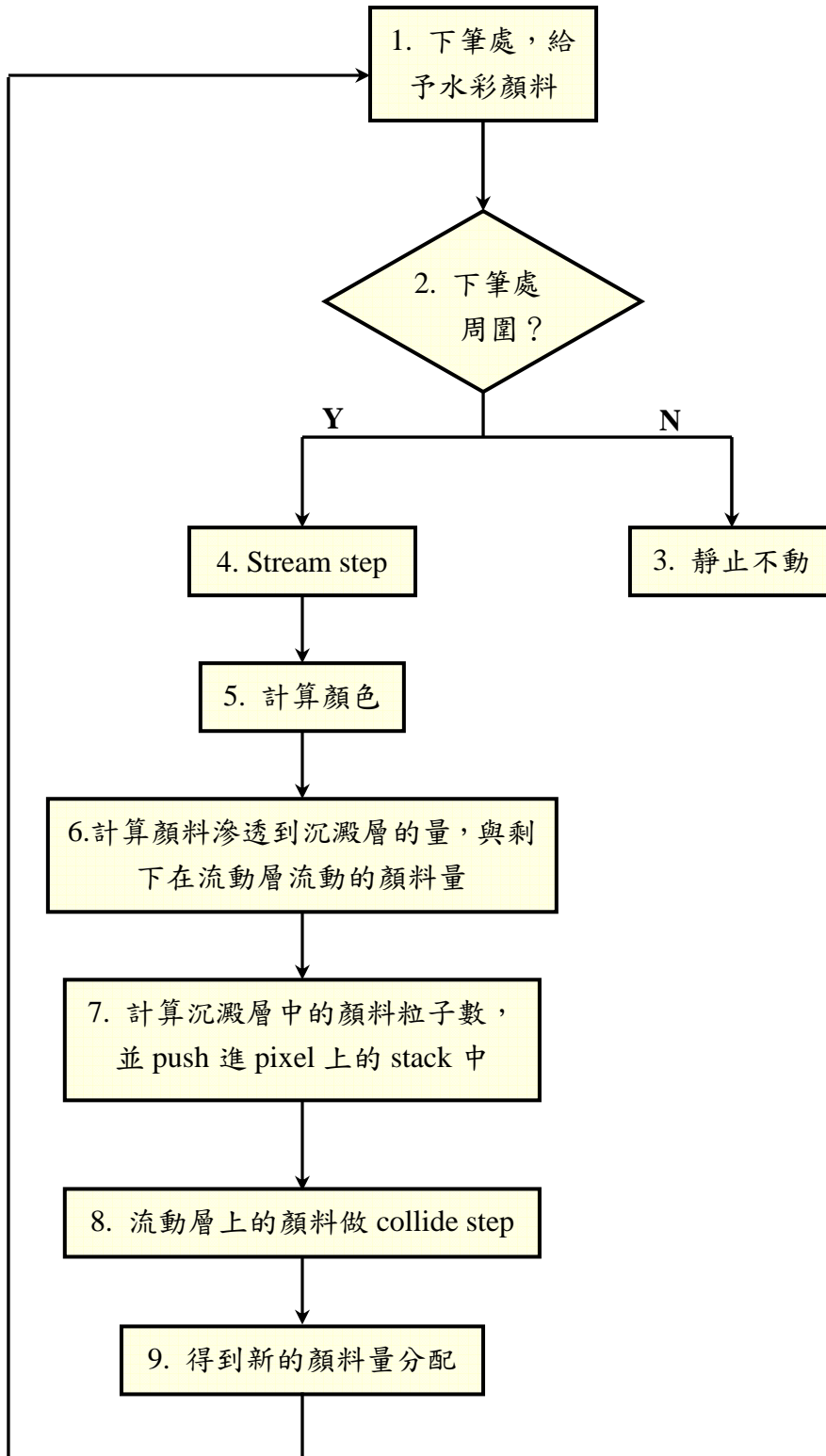
此水彩模擬在早期的系統中已有一個不錯的發展成果[32]，在本章節中，將介紹其水彩模擬流程。在此模擬中我們主要是利用 3D model-based NPR-System 為基礎，藉由流體力學中 Lattice Boltzmann Method(LBM)的導入，建構出水的模型，用來模擬水在畫紙上流動、擴散與其他的一些物理現象。然後，以這個水模型為基礎依照不同的水彩濃度，來產生出水彩多樣的色彩色調與渲染效果，進而模擬出水彩的繪畫風格。

在研究中，我們將虛擬畫布分成兩層，分別為「流動層」與「沉澱層」。一開始，水彩畫在虛擬畫布上，然後因為水彩中水的特性，水彩會在流動層中流動與擴散，漸漸的，水彩粒子會慢慢的在沉澱層中沉澱與累積。

所以，在我們的研究主要有兩個模擬部份，分別為水的擴散與顏料粒子的沉澱。

5.2 水彩模擬流程圖

水的擴散，是隨著時間的遞增向外擴散的，所以每一個時間(t)的運動都必須參考上一時間($t-1$)的擴散結果再來做向外擴散的運動，之後再依此運作不斷擴散出去，而顏料粒子是溶解在水中的，所以顏料粒子會隨著水在畫紙上流動並沉澱。[圖 22]為此研究的水彩模擬流程圖：



[圖 22] 水彩模擬流程圖

以下將說明水彩模擬流程圖的各個步驟：

1. 下筆處，給予水彩顏料

每一次都去判斷，筆觸落筆會落在哪些 pixel 上，並給予落筆處的 pixel 水彩顏料來源。

2. 下筆處周圍擴散作用

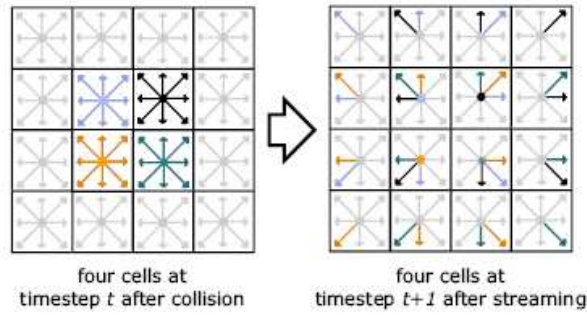
用來判斷 pixel 是否在下筆的筆觸周圍。因為我們假定當水彩剛畫在畫布上時，並不會馬上滲透進畫布中，水彩中水的流動特性，會對周圍的 pixel 造成某個程度上的影響。例如：以下筆處為中心，給一個 10*10 的區塊，則在這 10*10 區塊上若 pixel 上有水彩顏料便會作水的擴散模擬。

3. 靜止不動

若該 pixel 不在下筆的筆觸周圍，則此 pixel 不受影響，pixel 不會得到任何顏料且 pixel 上的顏料靜止不動。

4. Stream step

若屬於剛下筆的 pixel 或其周圍的 pixel，此時 pixel 會做 LBM 的 stream step simulation。將 pixel 中的 distribution functions 複製到相鄰 pixel 的相對向量上，再將 pixel 上每一向量上的數量加總起來，便會得到這個時間點上此 pixel 所有的顏料數了。



[圖 23] stream step simulation 前後圖。右圖為做完 stream step simulation 後的分配圖，再將每一 pixel 向量上的數量加總起來，便是此次的模擬所得到的顏料量了

5. 計算顏色

做完 stream step，pixel 需要計算從四面八方得到的顏料所混合出來的顏色。更詳細的顏色計算將於下一節 5.3 小節做介紹。

6. 計算顏料滲透到沉澱層的量，與剩下在流動層流動的顏料量

將虛擬畫布分為流動層與沉澱層，根據經驗法則，當顏料粒子越多時則沉澱越多，反之，則沉澱越少。當粒子沉澱在畫布上的越少，留在流動層上的粒子越多，顏料粒子會隨著水的擴散被帶的越遠，形成的筆觸顏色較淺且較不明顯；反之則顏色較深且筆觸明顯。目前在我們的系統中，給定一個使用者參數用來表示粒子沉澱的比例。



[圖 24]不同沉澱比例所造成的筆觸樣式。(a)1/100 (b)1/50 (c)1/10

7. 計算沉澱層中的顏料粒子數，並 push 進 pixel 上的 stack 中

對於每一次顏料的沉澱，我們認為顏料在畫布上是一次、一次由下而上累積的，所以我們提供虛擬畫布上的每一個 pixel 一個 stack 資料結構，用來存放每一次沉澱的顏料粒子。而 stack 「先進後出」的特性也符合了最先畫的顏料會在最下層，而最後畫的顏料在最上層的性质。

8. 流動層上的顏料做 collide step

在 collide step 會使用流動層上所剩的顏料來做計算，利用 pixel 上的 distribution functions 計算出 ρ 與 \bar{u} ，並代以下二個式子

$$f_i^{(0)} = w_i \left\{ \rho + \rho_0 \left[3(\bar{e}_i \cdot \bar{u}) + \frac{9}{2}(\bar{e}_i \cdot \bar{u})^2 - \frac{3}{2}(\bar{u} \cdot \bar{u}) \right] \right\}$$

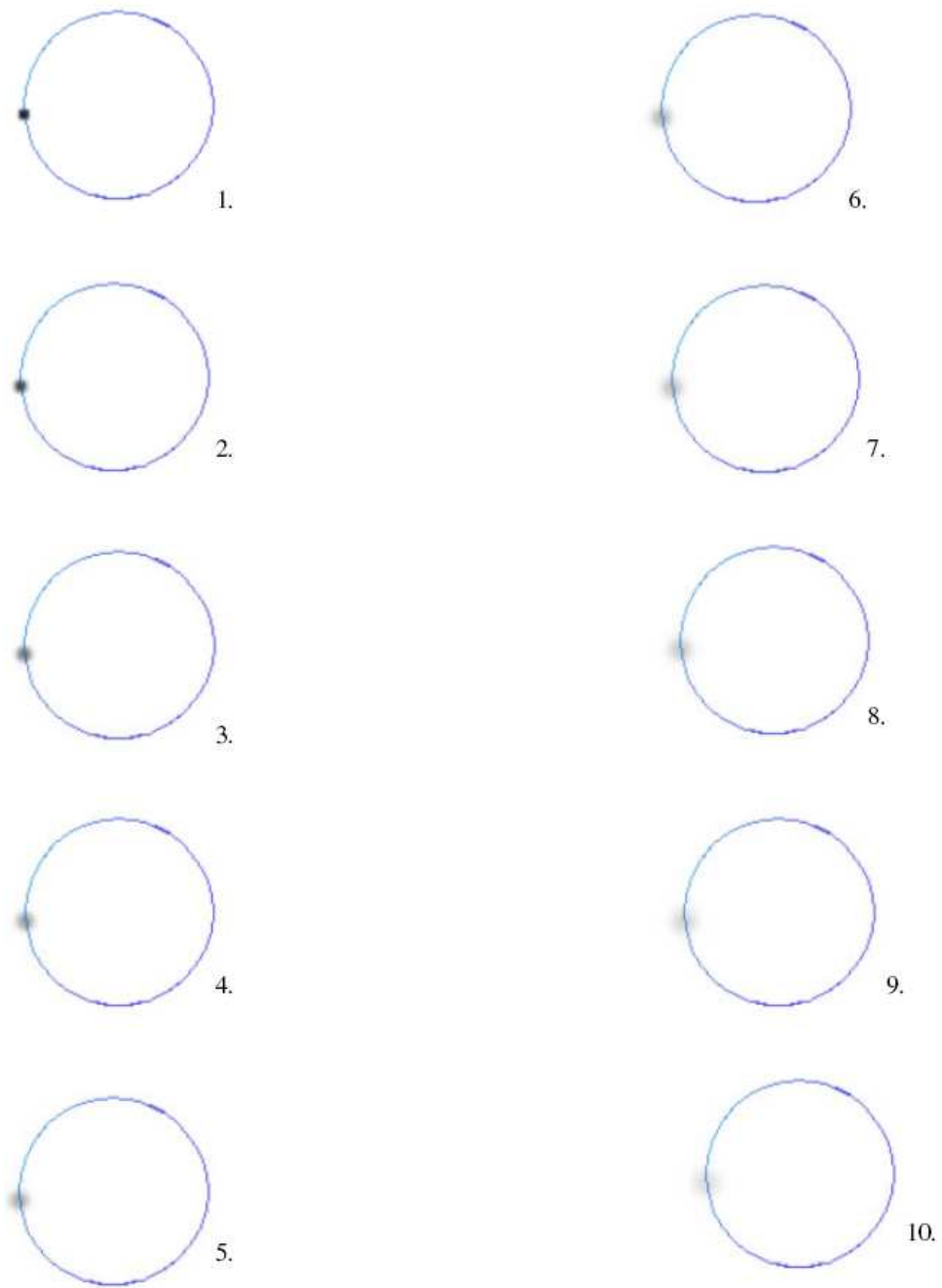
$$f_i' = (1 - \omega)f_i + \omega f_i^{(0)}$$

便可以得到新的顏料粒子分配。

9. 得到新的顏料量分配

得到了新的顏料粒分配後，就儲存下來，以便下一時間的擴散使用。

之後，每當有水彩顏料畫在畫布上，就執行這個水彩模擬流程，直到最後一筆筆觸畫完。



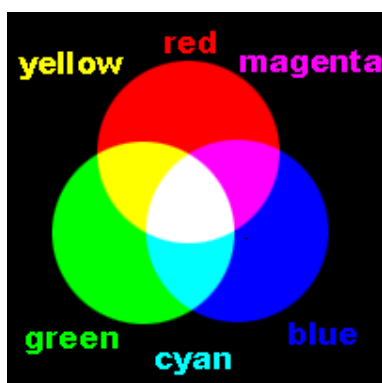
[圖 25]一滴水在虛擬畫布上的擴散過程



[圖 26]一筆水彩筆觸的繪畫過程

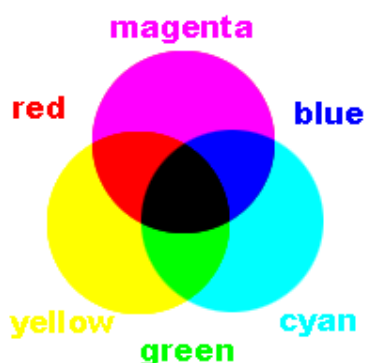
5.3 color blending

電腦裡與螢幕都是使用光的三原色(RGB)來標示顏色，也就是紅(red)、綠(green)、藍(blue)。RGB 使用的是「加色法」，三種顏色等量相加可以得到白色，兩兩等量相加可以得到中間色：黃(yellow)、青(cyan)、洋紅(magenta)。



[圖 27]RGB 色彩系統

但是在水彩的色彩系統中是採用「減色法」，三原色相加會得到黑色，這與 RGB 所採用的加色法不同，於是我們的系統使用了印刷上的 CMYK 系統，也就是青、洋紅、黃、黑。事實上除了黑色外，用顏料三原色無法混和出許多暗色系的顏色，為了彌補這個缺點，因此實際印刷的時候會額外加入黑色的顏料，以解決無法產生黑色的問題。因此就有所謂 CMYK 的色彩模式，K 表示黑色。



[圖 28]CMYK 色彩系統

因為顏料與 RGB 所採的色彩模式不同，所以我們在系統運作中必須要做色彩轉換的步驟。

首先，將 3D-model 載入到我們的系統中，得到的像素色是採用 RGB 計算的，但是因為水彩的顏料色彩模式與 RGB 不同，所以我們必須把得到的 RGB 值轉換成 CMYK 值，之後，在我們的系統中所有的色彩計算，都採用 CMYK 模式來做運算，等到最後要輸出到螢幕時，再將最終得到的 CMYK 值轉換成 RGB 作輸出，如此便可以在螢幕上得到我們水彩模擬的成果了。以下[圖 29]為利用 CMYK 三原色(青、洋紅、黃)兩兩相混所得到的二次色：

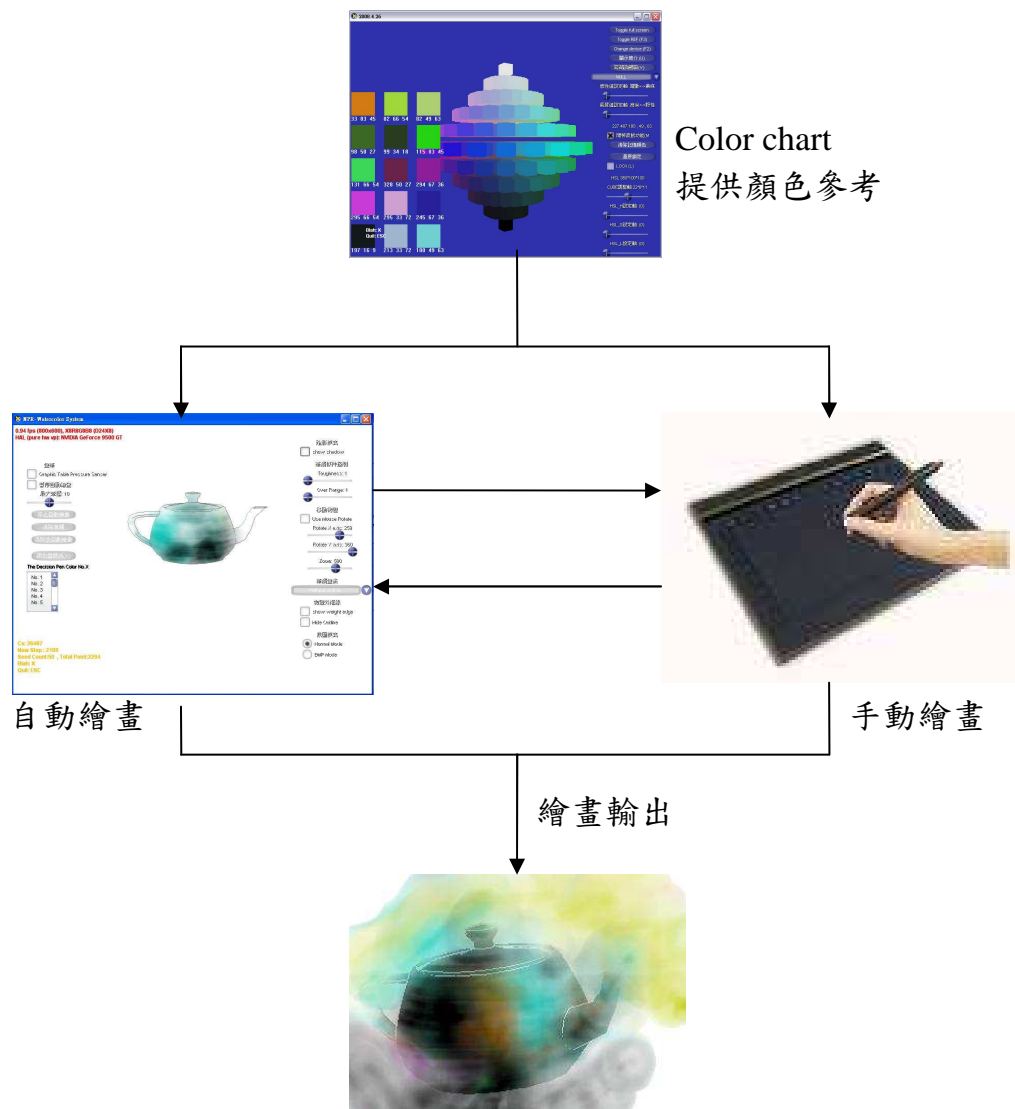


[圖 29]CMYK 三原色兩兩混色圖。(a)洋紅加青色得到藍色(b)青色加黃色得到綠色(c)洋紅加黃色得到紅色

第六章 GUI 設計

6.1 System overview

此 GUI 的設計最主要是以第四章所提到的 NPR 系統為基礎，整合繪圖板以及一個能夠提供客製化的 Color Chart 色彩系統[31]。使用者能夠搭配自動水彩筆觸以及手動水彩筆觸在虛擬畫布上面進行水彩繪畫模擬，並且能以 Color Chart 以做為顏色選擇的參考。以下[圖 30]為 GUI 設計概括流程：

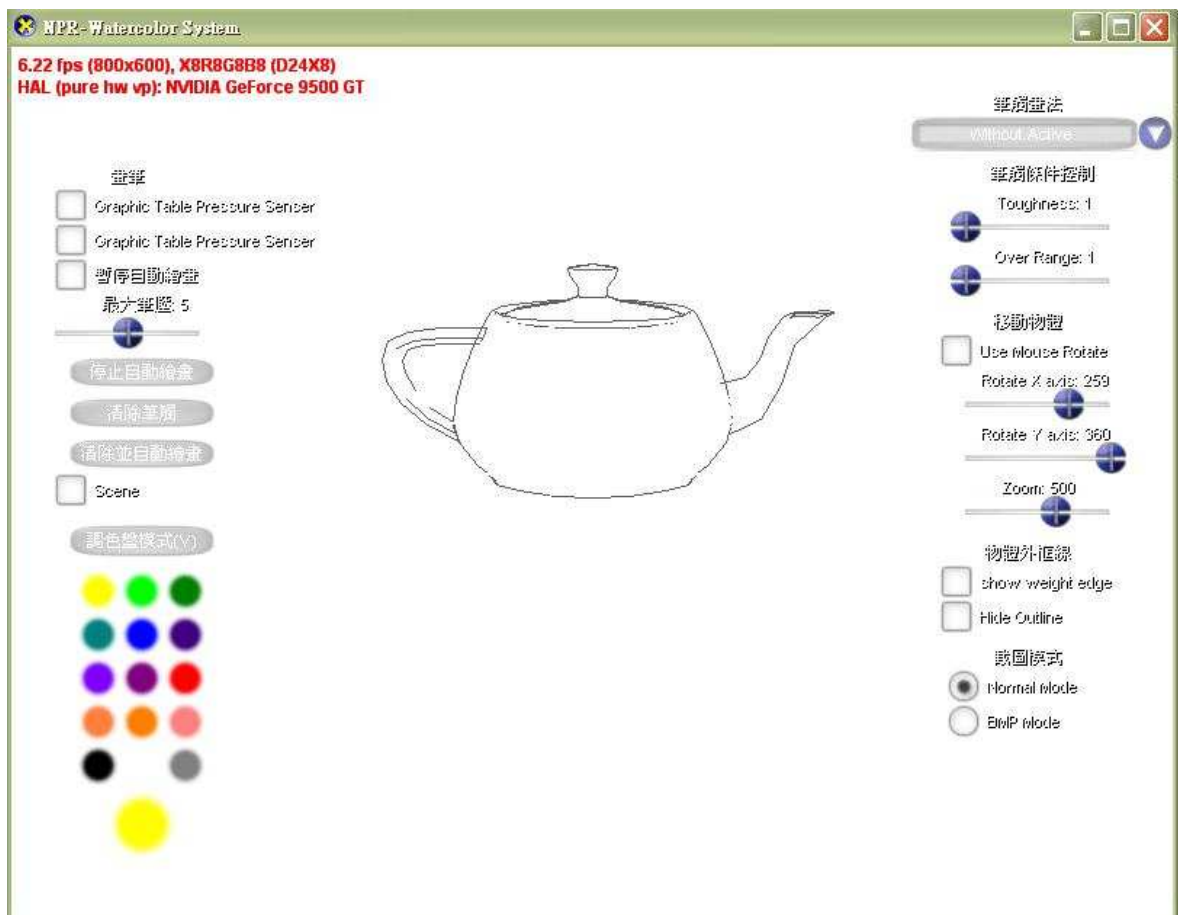


[圖 30] GUI 設計流程圖

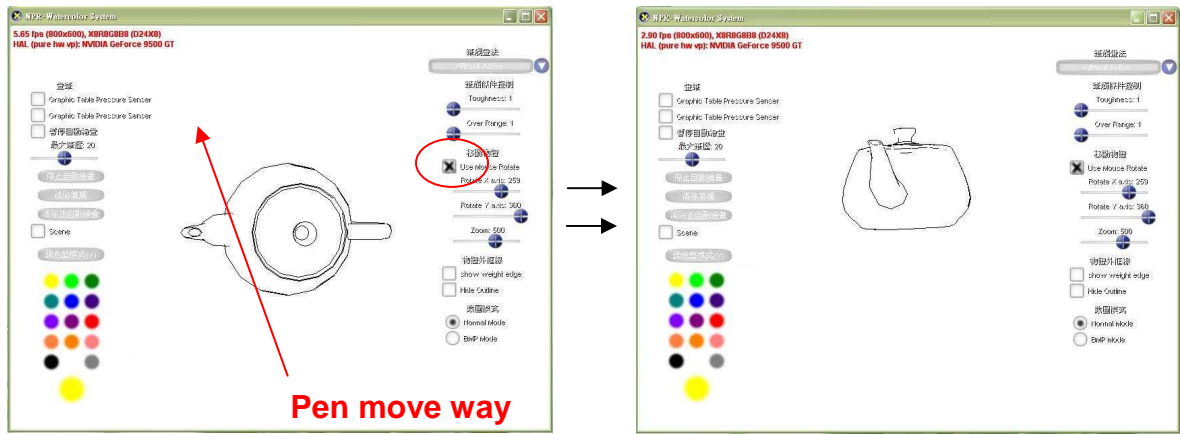
6.2 繪圖板整合

在此 GUI 中，我們整合繪圖板，提供使用者一個更直覺的方式進行手繪以及物體方向調整。

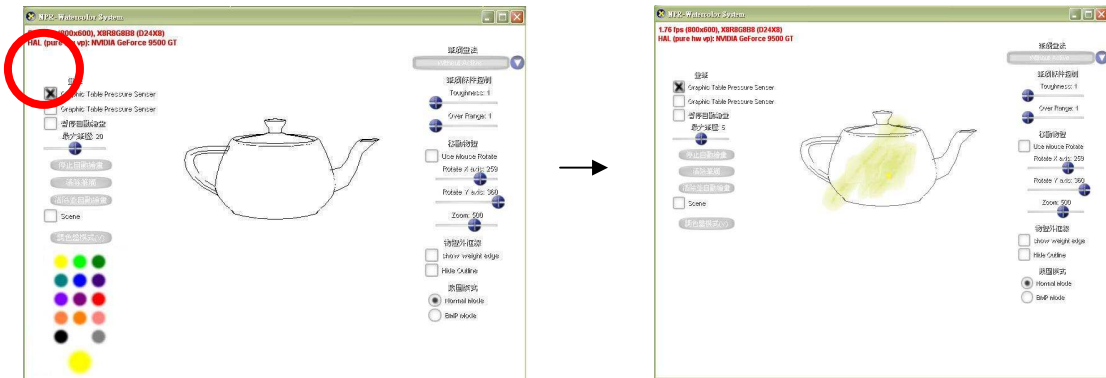
在使用時，繪圖板可以當作普通的滑鼠來點選 GUI 上的選項。[圖 31]為我們所提供的 GUI 介面。使用者可以用畫筆改變物體的方向[圖 32]，或以提供的顏色來進行手繪水彩[圖 33]。



[圖 31] GUI 介面



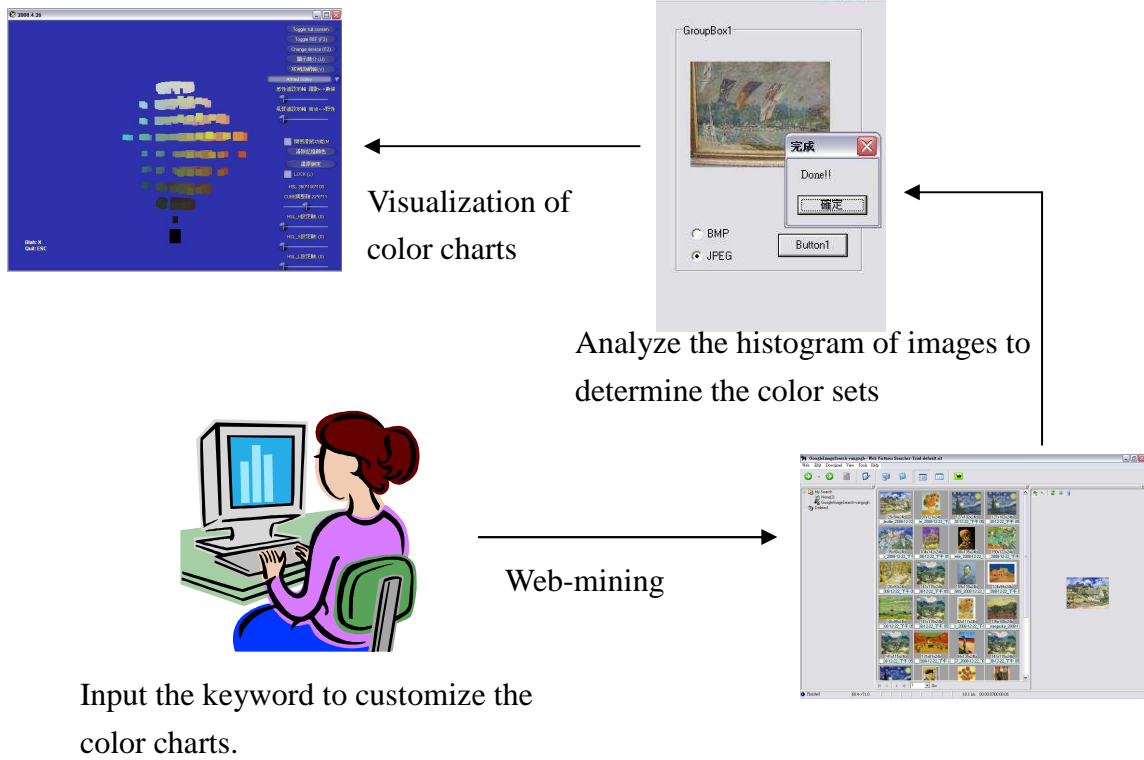
[圖 32]使用畫筆調整 Model 的方向。



[圖 33]使用提供的調色盤進行手繪。

6.3 Color Chart 系統整合

提供參考顏色的系統，我們採用 Color chart system[31]。此系統能夠讓使用者以 Web-mining 的技術，直接輸入藝術家姓名或其他關鍵字，在網路上搜尋相關圖片，再以圖片抽出色票資料，建構成一個立體的 Color Chart，提供使用者使用。[圖 34] 為 Color Chart 系統流程圖。

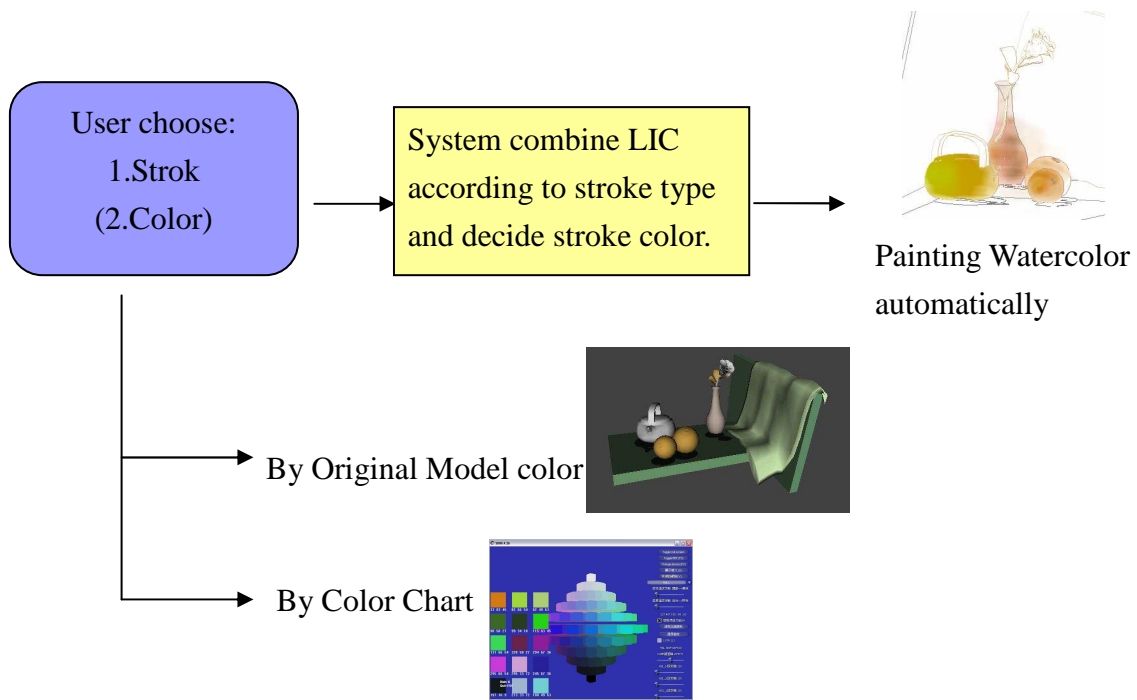


[圖 34] Color Chart 系統流程圖

第七章 筆觸模擬

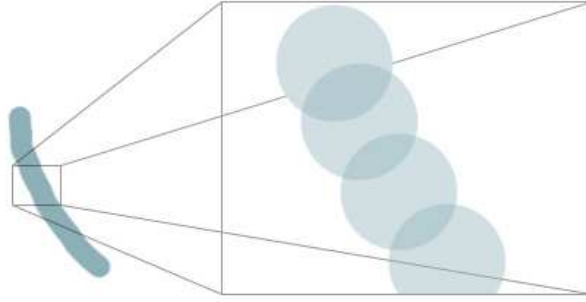
7.1 自動筆觸模擬

自動筆觸生成的部份，因為早期的水彩繪畫系統已有一個不錯的開發成果，所以在本研究中，我們只對此研究作些微的更動。[圖 34]為自動筆觸生成的流程圖。



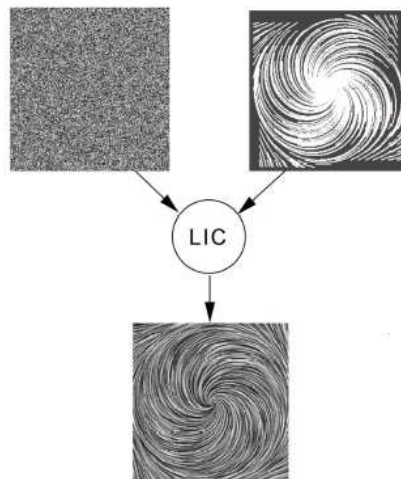
[圖 35]自動筆觸生成流程

在早期構想中，考慮最原始的筆觸構成。假設單點的筆觸圖案為一實心同色圓，則沿某一軌跡連續貼出此圖案即可模擬出最簡單的筆觸[圖 36]。而根據此一基本想法加以擴充，便產生了構成不同風格的筆觸所需的各種參數。



[圖 36] 筆觸的細節

而其筆觸效果是使用 Line Integral Convolution(LIC)來產生。簡單的 LIC 由一張 color/noise image 以及一張 Vector field 組成。對於 image 上的每一個 pixel，LIC 根據 vector field 追蹤一定長度的 pixel 並紀錄成一組組的 group，這一組組的 group 再經過顏色的混合運算，填色後，便能夠形成具有流紋效果的結果影像[圖 37]。再利用這種流紋般的特性，來生成筆觸的效果。



[圖 37] LIC 示意圖。Noise image 與 Vector field 輸入 LIC 的運算單元，生成具有流紋形的結果影像。

為了能使生成的影像更接近於普通的水彩繪畫，我們歸納出兩大類重要的筆觸，這些筆觸各自有著不同的走向特性、不同的粗細特性等。

7.1.1 2D Shape-based Stroke

在繪圖的時候，有時我們需要額外的線條來強調物體本身 2D 的形狀。大致來說，此種繪線常會隨著物件本身的 Outline 起伏、生成。為了能夠模擬出具有此種特性的繪線。首先我們必須決定存在於虛擬畫布上的每一個 pixel，分別受到哪些 outline pixel 的影響。

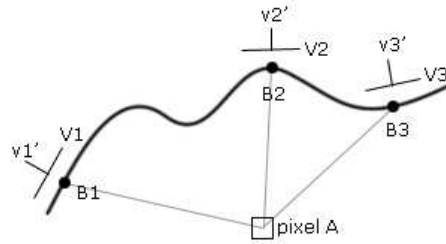
在處理這個步驟上，我們首先將所有的 outline pixel 以各自所屬的物件分類之，以加速影響筆觸的搜尋。同時，這個分類的步驟也可以避免不同物件間，不必要的筆觸相互影響。故，對於虛擬畫布上的每一個存在的 pixel，我們反覆執行以下的步驟：

(A)搜尋所有影響到該點的 outline pixel。

(B)給定一 threshold，判斷每一個 outline pixel 到該 pixel 的長度。若小於 threshold 則予以保留為 intensity pixel。

(C)根據篩選出來的 intensity pixel，作最後加總平均的計算。

所有的 outline vector 都是雙向的。若直接將所有找出的 intensity pixel 直接加總，則無法統一所有筆觸的方向。因此，相較於加總所有 intensity pixel 上的方向向量，我們給定一個向量位於 intensity pixel 且垂直於該 pixel 上的方向向量，然後加總這個向量。如[圖 38]所示。



[圖 38] 2D shape-based stroke vector 存取示意圖。Pixel A 為運算中的向量；B1,B2,B3 為範圍內的 Intensity Pixel；相較於加總 $V1, V2, V3$ ，我們加總 $V1', V2', V3'$ 三個可以確定其方向的向量。因為 V' 的方向可以根據作用中的 pixel 與其之間的夾角決定之。

7.1.2 3D model-based Stroke

當物件形體具有強烈的形狀特徵時，我們會依據模型本身的三維結構來給定其筆觸方向。

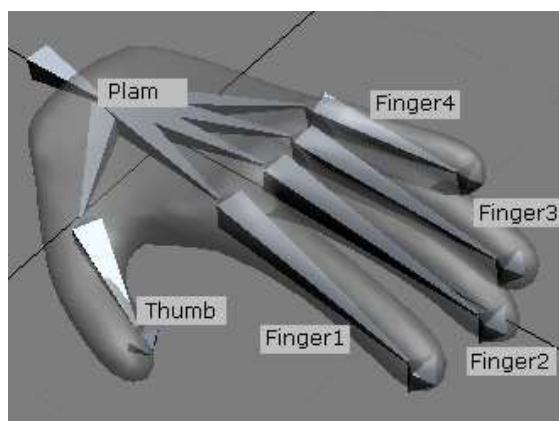


[圖 39] 柱狀體上的筆觸方向，為建模軟體中的基本型體之一。通常我們使用如圖上所示的方向筆觸來加強觀者對物件型體的認知。

在傳統的繪畫中，物件的形體表現一直是重要的一環。而筆觸在這個環節所扮演的角色，往往是用來強調出物件的三維特性，讓觀者能由 2D 的繪畫在腦海中重建出物件原始的 3D 型態。即是說，筆觸往往能夠幫助觀者更正確的認知一幅畫中的物件。

為了模擬出這類型的筆觸，我們採用模型本身具有的骨架結構。從模型中抽取出的模型建構法，由目前市面上較著名的建模軟體，如 3DS-MAX、MAYA 等所提供的骨架結構抽取之。

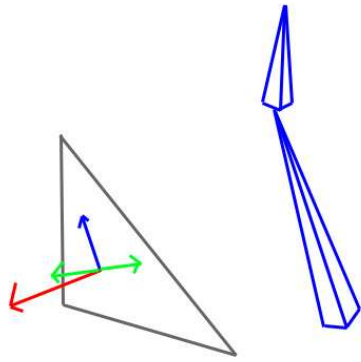
基本的骨架架構，由 Parent/Child 的階層結構組成。如[圖 40]所示，palm 為 thumb 以及其餘四個 finger 的父階層。而每一個 Bone 會有其影響的範圍、權重以及轉置矩陣，骨架依據這些可取得的資訊來控制模型的動作。



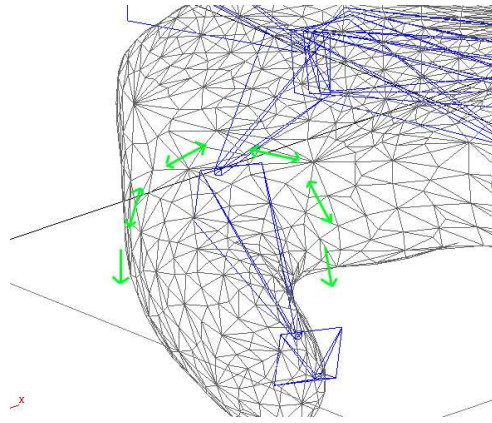
[圖 40] 手掌的骨架階層構造

我們從中取得骨架的相關資訊，即其個別的階層結構、範圍、權重以及轉置矩陣。以這些資訊來運算系統中的筆觸走向。

至此，根據每個 Bone 的影響範圍，我們可以讓每一個 Triangle 結構獲得一個 Bone 可供參考方向。外積此 Bone 方向向量與 Triangle 的法向量之後，可得到一個三維雙向性的筆觸方向，有了三維的方向向量之後，我們再對其作一次平面投影的矩陣運算後便可以獲得我們在 Buffer 上需要的二維方向向量。如[圖 41]、[圖 42]所示。



[圖 41] 3D model-based stroke vector 存取示意圖。藍色箭頭代表骨架方向；紅色箭頭為 Triangle 法向量。綠色為兩者



[圖 42] 手掌上，Thumb 部分的三維筆觸分布。

7.1.3 Specular

明暗度是水彩繪畫上一個重要的環節。在繪圖時常需要考慮物體上的明暗層次，以便做出物體的立體感。所以，為了使我們的水彩模擬更加豐富，我們參考了場景中給定的照明條件，在物體上產生 specular 的效果。Phong Shading 是將 Gouraud Shading 加以改進，是一種可以在物體上顯示出 specular 效果的 shading 方式，但是由於 Phong Shading 需要較長的運算時間，所以我們將 Phong Shading 的觀念帶入到 Gouraud Shading 中，最後，利用 Phong's reflection model 來計算出每一個 pixel 的顏色。

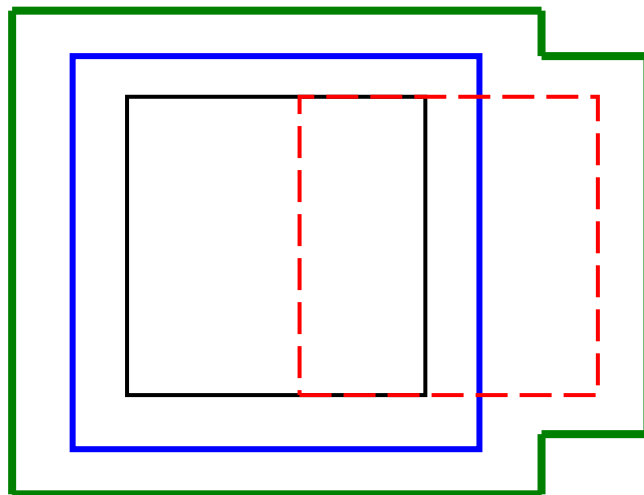
Phong Shading 是利用三角形頂點上的法向量去內插計算出表面上每一個 pixel 的方向向量，來得到每一個 pixel 的亮度。而在我們原本的 shading 中，我們增加對於三角形三個頂點上的亮度計算，再去做 Gouraud Shading 的計算，如此我們可以利用較少的時間得到類似 Phong Shading 的 specular 效果。以下[圖 43]即我們得到具有 specular 效果的影像：



[圖 43] 具有 specular 效果的影像

7.1.4 描繪過程

筆觸的生成是在每個時間 t 中會有一個新的筆觸位置下筆，而且同一筆筆觸的下筆都是具有連續性與相關性的。所以，必須要考慮到下筆處周圍 pixel 的擴散現象。例如：當筆觸的第一個下筆處下筆會擴散一次，隨著時間遞增，當第二個下筆處下筆，第一個下筆處應該要擴散第二次，而第二個下筆處擴散第一次，此後，依此類推。如 [圖 44] 所示。

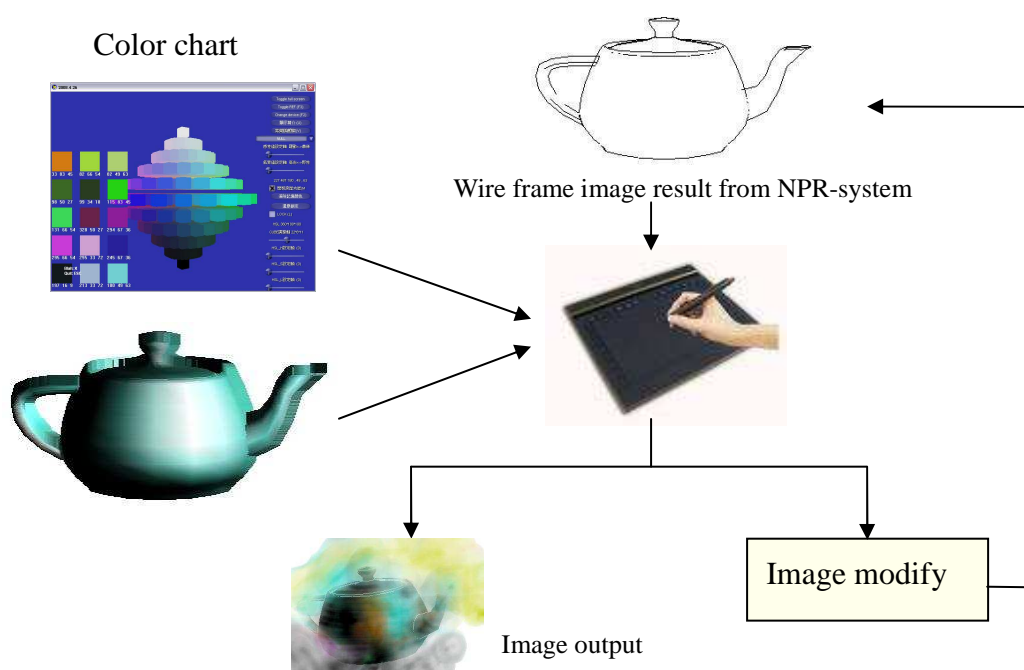


[圖 44] 水彩描繪過程示意圖。黑色框為第一個下筆處，第一個下筆處會擴散成為藍色框。隨後，紅色框表示第二個下筆處加入，而藍色框會與紅色框一起擴散成為綠色框。

而在筆觸的色彩上，若選用參考 model 上的顏色為筆觸色彩，因為 model 是依照場景光源位置而反映明暗效果的，而且每一條筆觸是由多個 pixel 組成，所以在筆觸的色彩生成上，可以產生具有明暗效果的筆觸。這個方法可以不需要使用者的指定輸入，我們就可以自動化的產生具有明暗效果的筆觸。若選用 Color Chart 為參考顏色，則其顏色也會因受到 Model 上光線的明暗變化，而產生具有明暗效果的參考顏色。

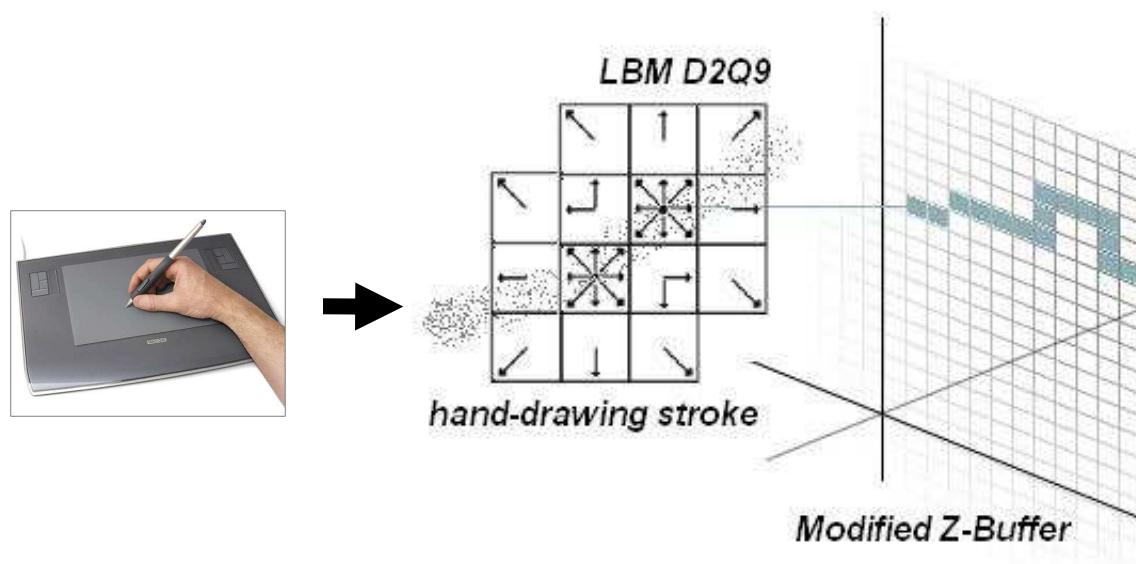
7.2 手繪筆觸模擬

相對於自動筆觸的部份，我們提供一個能夠以較直覺的繪圖板方式來進行手繪水彩。[圖 45]為手繪筆觸模擬的流程圖。我們以自動筆觸模擬的概念來假設筆觸模擬的繪圖軌跡。同樣也是在虛擬畫布上，但是筆觸則是使用以繪圖板上所得的使用者移動軌跡。而筆觸顏色則是提供兩種選擇，第一種以 Model 顏色為參考，但改以使用者手繪軌跡取代自動繪畫的軌跡，第二種則是讓使用者自行選取顏色，在虛擬畫布上作畫。而使用顏色繪畫時，使用者也可以在繪畫的當下觀察顏色的混色。



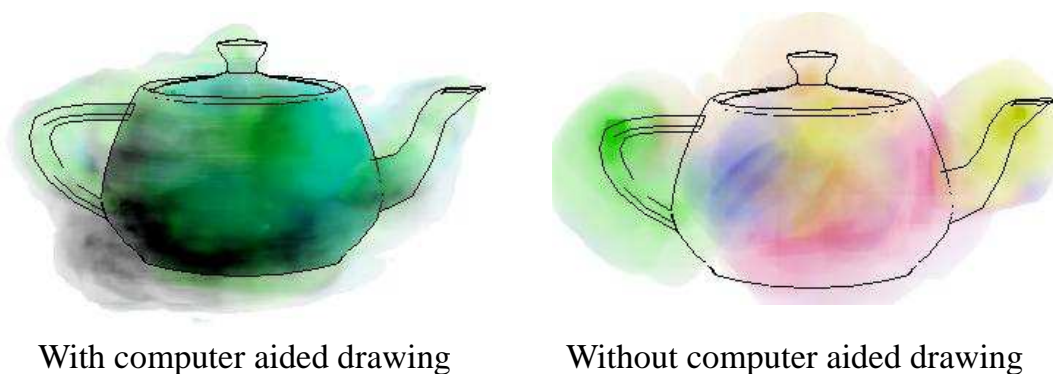
[圖 45]手繪筆觸模擬的流程圖

水彩的筆觸，我們也使用 LBM 為模擬水彩基礎，在下筆的同時進行水彩擴散的計算。計算之後，將結果儲存在 Z-buffer 中。而顏色的混色效果，亦是由 Z-buffer 中所儲存的顏色及水量當作沉澱層，與筆觸所給予的顏色做即時的混色[圖 46]。



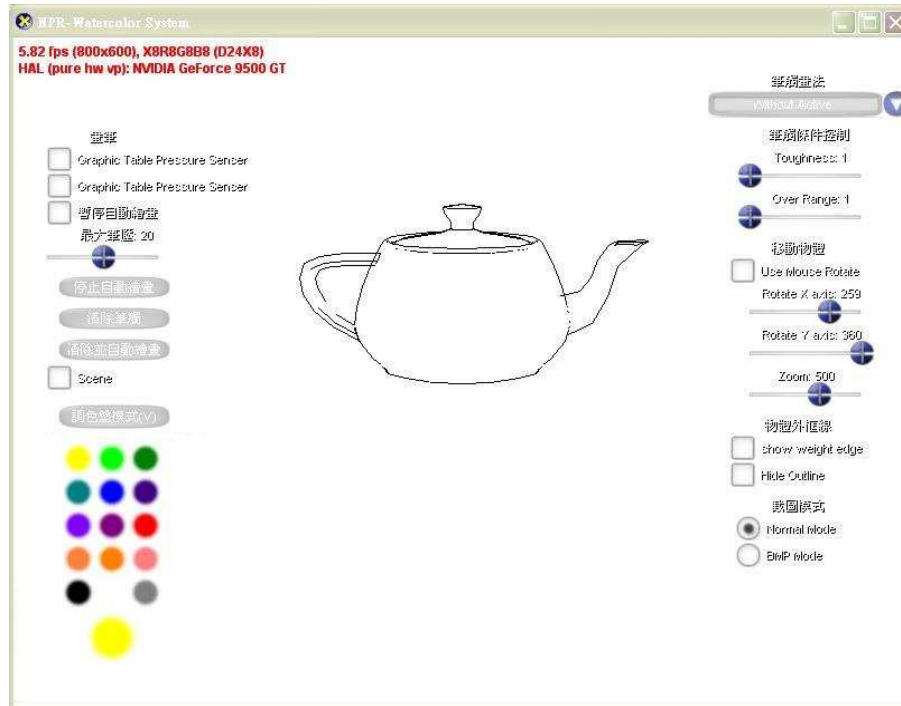
[圖 46] 手繪筆觸使用 LBM 在虛擬畫布上的模擬

使用者可以參考自動繪圖的色彩明暗，初學者亦可利用自動繪圖當作輔助，學習正確的水彩繪畫技巧。[圖 47] 為以自動繪圖為基底參考的水彩繪畫與直接使用線框架，沒有考慮到明暗度的水彩繪畫。

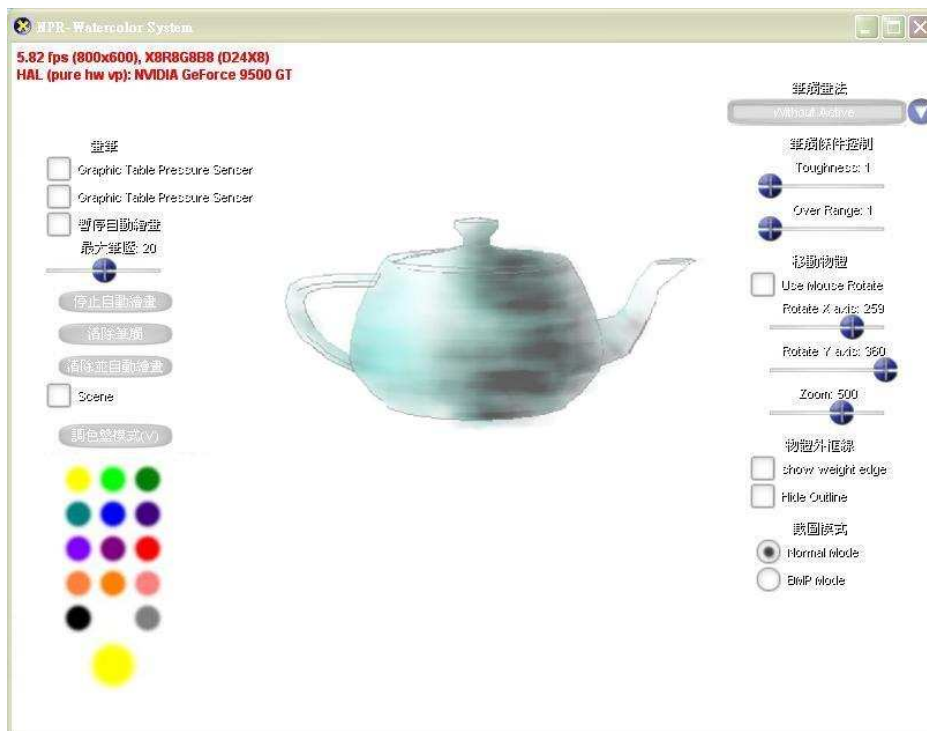


[圖 47] 左圖為採用自動繪畫為基底，有明暗度的水彩畫。右圖為任意的水彩繪畫。

第八章 研究成果



[圖 48] GUI 介面



[圖 49]使用自動繪畫的 GUI 介面



[圖 50] Teapot-描繪過程:自動繪圖 3Dmodel-based Stroke



[圖 51]線框架



[圖 52]在線框架直接繪圖的手繪水彩繪畫



[圖 53]以自動為基底的手繪水彩繪畫

更多的水彩風格影像與動畫，請參考實驗室網站

<http://cgda.csie.ncnu.edu.tw/gallery.htm>

第九章 結論以及未來工作

9.1 結論

在本篇論文中，我們實做了一個 NPR 水彩系統。我們載入 3D-model 當作原始資訊，並根據 model 與場景資料，能夠自動化的產生具有水彩繪畫風格的影像或動畫。

首先，從水彩的特性與技法來做分析與量化，希望能夠由水彩的本質：「色彩」、「綜合」與「光」為基礎，模擬出水彩繪畫風格。應用 Lattice Boltzmann Method 模擬流體，並探討水彩濃度與不同顏色顏料混合所造成色彩上的差異。

在 GUI 的設計上，我們結合了 NPR 水彩系統、Color Chart 系統及繪圖板。在自動繪圖方面，根據場景資訊並利用 Phong's reflection model 來計算出 model 的顏色，因為筆觸顏色是參考 model 上的顏色得來的，而每一條筆觸由多個 pixel 構成，隨著 pixel 的移動，在同一筆筆觸上就有了明暗度的變化。這個設計提供給水彩繪畫初學者一個很好的繪畫參考。本研究還探討了二種水彩筆觸繪畫手法，我們也從早期的研究中得到這兩種筆觸，分別為 2D shape-based Stroke 與 3D model-based Stroke。而手繪方面提供了一個更自由的顏色選擇方式，並讓使用者能在繪畫的同時，能同時觀察到色彩間的混色。

由我們的研究，或許無法造出可以取代傳統藝術畫作的成品，然而若能將此近似的水彩成像手法以電腦演算，並且加入使用者的創意，除了能夠提供初學者一個學習水彩繪畫技巧的平台，也能由不同的方向，給予藝術家一個不一樣的創作空間。

9.2 未來工作

本篇研究藉由Lattice Boltzmann Method來模擬流體在虛擬畫布上的流動，並由Modified Z-buffer與3D-model based的許多概念進行整合。目前只有針對兩種筆觸的走向以及特性做處理，在畫布上個別位置的溼度差異與光所造成的陰影效果，都尚欠缺考慮，另外，對於跟使用者互動的介面也可以加入系統中。

底下列出可在未來改進此系統的幾個方向：

1. 更多筆觸的開發

目前系統只考慮了 2D shape-based Stroke 與 3D model-based Stroke，還有已分析的 Feature- based Stroke 與 Texture- based Stroke 尚未加入本系統，我們相信還有許多筆觸特性是還沒考慮到的，希望藉由本系統，未來模擬出更多的水彩筆觸效果。

2. 在 NPR 作品的展現中，反映畫家作畫時可能採取的流程

目前系統筆觸的 seed 是以亂數灑點的方式生成，而畫家在作畫時，通常有一定的作畫流程，所以未來可以考慮將畫家實際作畫的流程整合進本系統的水彩模擬上。

3. 畫布上多種濕度的筆觸表現

我們的系統目前尚未考慮到畫布上的溼度，但是，在實際繪畫時，畫布不同位置的溼度是有差異的。

特別是，我們知道畫家會刻意去控制畫布上的溼度。例如，使用吹風機烘乾部分的區域，或將整幅畫晾乾兩天之後，再繼續創作，反之，也會大量刷濕特定

區域，再渲染以產生朦朧的效果。如何提供一個 GUI，讓使用者能夠以直觀、互動的方式，模擬上述的過程，也是我們未來思考的方向之一。所以，若能夠在我們系統上加入一個讓使用者自行決定濕度的介面，與一個能夠即時反應畫布上濕度的相對應視窗，讓使用者能夠掌握畫布上的溼度資訊，我們相信模擬出來的水彩風格影像會更有趣。

4. 陰影效果

我們的系統已經模擬了光對於水彩色彩的影響，但在於光所造成的陰影效果還欠缺考慮，所以，我們的系統未來考慮使用如：Stencil Buffer，來解決陰影的問題，包括畫布上的陰影、物體上的陰影與陰影重疊的問題，希望加入陰影後，會使的我們的系統更加完整。

5. 使用者介面的開發

我們希望提供一個 3D 的場景，這個 3D 場景可以讓使用者自由的配置那些經由與系統的互動所產生出來的水彩 3D-model，而這些 3D-model 也會因為在場景中擺放位置的不同，而呈現出不一樣的效果，並且這些效果都是遵從水彩畫裡 object 配置位置所應有的表現方式。我們期望能夠提供一個空間讓 user 發展自我的風格。

參考文獻

1. Tateosian, L. G. and Healey, C. G. "NPR: Art Enhancing Computer Graphics." Technical Report TR-2004-17 (2004), Department of Computer Science, North Carolina State University.
2. Haeberli P. "Paint by numbers: Abstract image representations." Computer Graphics, 1990, 24(4):207-214.
3. Georges Winkenbach and David H. Salesin. "Computer-generated pen-and-ink illustration." In Andrew Glassner, editor, SIGGRAPH 94 Conference Proceedings, pages 91–100, Orlando, Florida, 1994.
4. William Baxter , Jeremy Wendt , Ming C. Lin, "IMPaSTo: a realistic, interactive model for paint", Proceedings of the 3rd international symposium on Non-photorealistic animation and rendering, June 07-09, 2004, Annecy, France
5. Curtis, C. J., Anderson, S. E., Seims, J. E., Fleischer, K. W., And Salesin, D.H. "Computer-generated watercolor." SIGGRAPH 1997 Conference Proceedings. pp. 421-430. 1997.
6. Thomas Luft, Oliver Deussen, "Real-time watercolor illustrations of plants using a blurred depth test." NPAR 2006: 11-20
7. Johan, H., Hashimota, R., Nishita, T. 2005. "Creating watercolor style images taking into account painting techniques." Journal of Artsci, 207-215.
8. Bousseau, A., Neyret, F., Thollot, J., And Salesin, D. 2007. Video "Video Watercolorization using Bidirectional Texture Advection." In ACM Transactions on Graphics, 26(3): 104:1-104:7.
9. David Norman 著.羅若蘋 譯."繪畫大師Q&A·水彩篇",視傳文化事業有限公司。
10. Lieu-Hen Chen, Hsin-Hao Lin: Shape-Oriented Brush Stroke Synthesis in Non-Photorealistic Rendering. CGW2006
11. 謝明鋁 著, "水彩畫法的奧秘", 雄獅圖書股份有限公司, 1996初版
12. David L Small, "Simulating Watercolor by Modeling Diffusion, Pigment and Paper Fibers", SPIE Proceedings, Vol. 1460, No. 15. San Jose, CA, 1990.
13. T. Van Laerhoven, J. Liesenborgs, and F. Van Reeth. "Real-time watercolor painting on a distributed paper model." In Proceedings of Computer Graphics International 2004.
14. Nelson S.-H. Chu and C.-L. Tai. "Extending MoXi to simulate Western Watercolor." Department of Computer Science and Engineering, University of Science and Technology, Clearway Bay Kowloon, Hong Kong, 2008.

15. Thomas Luft, Oliver Deussen, 2005. "Interactive watercolor animations." In PG: Pacific Conference on Computer Graphics and Applications, 7-9.
16. Lum, E. B., And Ma, K.-L. 2001. "Non-photorealistic rendering using watercolor inspired textures and illumination." In Proceedings of Pacific Graphics '01, 322-330.
17. Burgess, J., Wyvill, G., And King, S. A. 2005. "A system for real-time watercolour rendering." In CGI: Computer Graphics International, 234-240.
18. Lei, E., And Chang, C.-F. 2004. Real-time rendering of watercolor effects for virtual environments. In PCM: Pacific Rim Conference on Multimedia, 474-481.
19. Thomas Luft, Oliver Deussen, 2005. "Real-Time Watercolor for Animation." Pacific Graphics 2005, Macau.
20. Jiajian Chen, Greg Turk and Blair MacIntyre. "Watercolor Inspired Non-Photorealistic Rendering for Augmented Reality".
21. James A. Liggett. "Basic equations of unsteady flow." Unsteady Flow in Open Channels, Vol. 1, eds: K. Mahmood and V. Yevjevich, Water Resources Publications, Fort Collins, Colorado, 1975.
22. C. B. Vreugdenhil. "Numerical Methods for Shallow-Water Flow." Kluwer Academic Publishers, 1994.
23. Bousseau, A., Kaplan, M., Thollot, J., And Sillion, F. X. "Interactive watercolor rendering with temporal coherence and abstraction." In International Symposium on Non-Photorealistic Animation and Rendering (NPAR), 2006, 141-149.
24. Jos Stam. "Real-time fluid dynamics for games." In Proceedings of the Game Developer, mar 2003.
25. Nelson S.-H. Chu, Chiew-Lan Tai. "Moxi: real-time ink dispersion in absorbent paper." In Siggraph 05, ACM Press, 2005, 504-511.
26. Chet S. Haase and Gary W. Meyer. "Modeling pigmented materials for realistic image synthesis." ACM Trans. on Graphics, 11(4):305, October 1992.
27. G. Kortum. "Reflectance Spectroscopy." Springer-Verlag, 1969.
28. Yu D., Mei R., Luo L-S, Shyy W. "Viscous flow computations with the method of lattice Boltzmann equation." Progress in Aerospace Sciences, 2003.39,329-367
29. Nelson S.-H. Chu, C-L Tai. "MoXi: Real-Time Ink Dispersion in Absorbent Paper."
30. THUERNEY N. "A single-phase free-surface lattice-Boltzmann method.", 2003
31. L.H Chin, Ting-Yu Liu. "3D visualization of color charts basis on the web mining technology", 2009.
32. Lieu-Hen Chen, Yi-Jing Lee, Yi-Hsin Ho: Simulating NPR Watercolor Painting based on the Lattice Boltzmann Method. CGW2008.