

## 可適應使用者操控行為的智慧型 3D 介面

許書璋  
國立政治大學  
資訊科學系  
s8810@cs.nccu.edu.tw

李蔡彥  
國立政治大學  
資訊科學系  
li@nccu.edu.tw

### 摘要

在 3D 虛擬環境的操控介面中, Walk 模式是最常用的瀏覽模式之一;但在使用 Walk 模式瀏覽場景時,使用者的視角比真實世界來的小,而且沒有視角餘光的輔助,導致在操作過程中常會因身體兩側碰撞到場景中的障礙物而有停滯不前的現象。為了改善這個問題,文獻中已有研究提出以運動計劃與虛擬力場等的瀏覽輔助方式,減少碰撞的機會,增加使用者的瀏覽效率。但目前這些輔助方法,尚無根據不同使用者的個人特性動態調整輔助程度的機制。本研究即是在過去研究成果的基礎上,嘗試提供個人化的 3D 場景瀏覽輔助機制。我們設計了兩種方法:模擬實驗和動態調整,分別以離線(off-line)及線上(on-line)的方式,針對不同使用者對輔助機制的反應,找出對該位使用者最佳的虛擬力場輔助參數,讓虛擬力場輔助的機制達到最佳的效率。從實驗結果中可以發現,兩種方法都能夠進一步提升使用者操控虛擬環境的效率。

### 1. 簡介

虛擬實境是電腦模擬出來的互動式三度空間環境;使用者透過 3D 人機介面體驗擬真的虛擬環境。虛擬實境的應用,在電腦、工業、航太、醫藥、教學、軍事等領域都十分廣泛,例如電腦輔助設計、機械人輔助設計、飛行模擬、虛擬博物館、分子結構分析等。由於電腦傳統輸入及顯示介面的限制,虛擬實境系統的人機介面設計變得相當重要。一個設計良好的 3D 人機介面,可以幫助使用者更容易且更有效率的操控虛擬場景,使他們能夠更專注於他們想完成的工作。

目前在全球資訊網(WWW)上,使用者只要透過 VRML 瀏覽器(VRML Browser),就可以瀏覽由 VRML 所建構出的虛擬場景。一般 VRML 瀏覽器所提供的瀏覽模式(navigation modes)包含 Walk、Pan、Turn、Roll 等。其中 Walk 模式應是最常使用的瀏覽方式了。使用者透過此模式可以前進、後退、及左右旋轉等方式在 3D 場景中遊走(Walkthrough)。不過使用 Walk 模式瀏覽 3D 場景時,由於視角大小的限制及缺乏餘光的輔助,常會因碰撞到場景中的障礙物,而導致使用者在場景中前後嘗試但仍然停滯不前。為了改善這個操作的問題,在文獻中曾有研究以機器人學中的運動計劃演算法[4][7],或以虛擬力場[6]等方式,輔助使用

者以順暢的方式通過場景中的困難地帶。

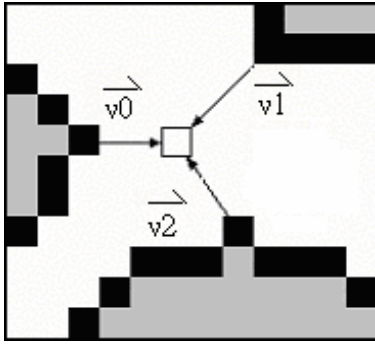
然而,由於使用者對 3D 場景的熟悉度不一,或是對 3D 瀏覽器操控介面的熟悉度,都會影響到使用者在 3D 場景中的操控行為。這時候輔助瀏覽的程度,就應該依照使用者的特質而有所改變。因為如果對一個非常熟悉此場景的使用者給予太多的輔助,有時反而會限制了他的活動。若對不熟悉場景或不大會操作介面的使用者給予的輔助太少,可能還是會覺得操作上的不方便。本文所探討的問題就在於如何提供一個適性的智慧型 3D 介面,根據使用者在目前的虛擬力場輔助下的表現,調整虛擬力場的參數組態,使虛擬力場達到最適合該位使用者的操作行為,進而提升虛擬實境中操控效率。

### 2. 相關研究:

3D 人機介面是使用者和虛擬環境溝通的管道。在 3D 人機介面設計的研究方面,因為有許多不同的虛擬環境應用,衍生出許多不同的類型的設計,如頭帶顯示器(Head Mounted Display)、資料手套(data gloves)、3D 軌跡追蹤裝置(3D tracking devices)、力回饋控制器(force feedback joysticks)等這類的虛擬實境類的 3D 人機介面裝置。而以一般個人電腦及 2D 滑鼠操控 3D 虛擬環境[11][12],可以讓更多電腦使用者接觸到 3D 虛擬環境所能提供的應用。但是因為使用低維度的裝置在高維度的空間操作,會有操作上不自然的問題產生,導致操作困難度提升,操作效率降低。為了解決這個問題,有許多提升虛擬環境操控效率的相關研究被提出來[6][7][8]。

以人工智慧技術建立 3D 瀏覽輔助介面的研究包含以運動計畫器[7]及虛擬力場[6]的方式提高執行效率。所謂利用運動計劃(motion planning)來輔助 3D 瀏覽,就是在使用者的操作過程中,如果使用者的目標點會與某個物體碰撞,就需要適當的修改。當他目前的位置與目標點之間有物體阻擋時,就用運動計劃的方法,繞過這個物體。這不只是能夠在碰撞之前先預防,在碰撞之後也還有機會能繞過物體。

虛擬力場是由虛擬空間中的物體邊緣產生排斥力所形成的。如果讓使用者一直受到來自於物體邊緣的排斥力的影響,使他每一次的移動,都會比原本沒有排斥力影響時更遠離附近的物體,就可



圖一：自由空間裡的一點(用方框表示)，以向四周一個大小為 7x7 的區域，作為它的附近地區。以本圖為例，在這個區域內找到三個邊緣點，從這三個邊緣點指向中間自由空間點箭頭表示向量  $v_0, v_1, v_2$ 。各別計算其影響力，則該點最後的力則是三個向量的合力。

望達到減少碰撞次數的目的。減少碰撞次數，自然能夠讓使用者瀏覽時感覺比較順暢。

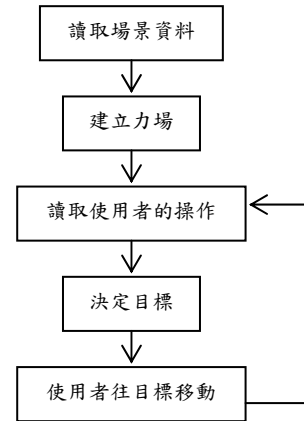
3D 人機介面的客制化也是 3D 人機介面研究很重要的一部份，客制化的機制能夠幫助讓使用者更容易或是更有效率的操作，[5] 的研究及是在虛擬環境中，以加強學習(reinforcement learning)的方式來實現使用者在虛擬環境中各種物體操控的客制化機制。

### 3. 虛擬力場運作方式

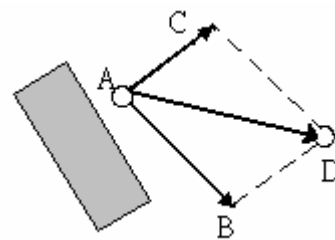
#### 3.1. 力場的建立與運作

我們在使用者可以自由運動的空間(free space)裡，對於每個位置(每一個點)，計算出它受到來自於物體的排斥力(力的大小與物體和使用者之間的距離成反比)(如圖一)，形成虛擬力場(force field)。使用者在瀏覽虛擬場景時，就根據這個力場，來決定他在不同的位置所受到影響力的大小與方向，輔助使用者的瀏覽。建立力場可以預先處理(preprocessing)的方式進行。建好的力場是一個二維的表格。使用者瀏覽時，在程式中直接查表，讀出使用者所在位置的力場，以減少計算量。

力場並不是直接對使用者造成影響，而是根據使用者的操作，配合力場中的力取出來作一些動態的調整後，才用來影響使用者。但是力場本身並沒有任何改變，所以它是一個靜止場。力場的運作流程如下：系統先讀取場景，根據場景資料建立力場後，再由瀏覽器讀入使用者對滑鼠的操作，瀏覽器會根據這個操作預定一個目標點，由使用者的操作行為與目前位置決定力場的影響，再依照影響力改變原本的目標點；使用者在下一個時間就會往這個新的目標移動。圖二為力場系統簡單的運作過程。圖三中，灰色矩形為一障礙物，A 點表示使用



圖二：力場運作流程圖



圖三：決定目標點

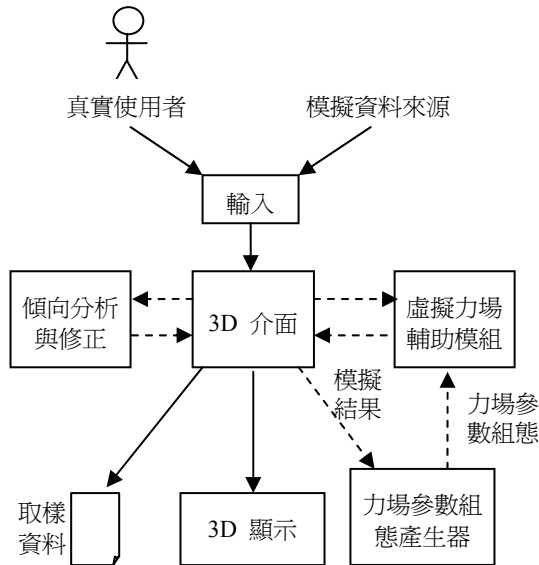
者現在的位置，向量  $AB$  是使用者輸入的操控向量，而向量  $AC$  是力場計算出來的影響力。D 點是 A 點的位置加上向量  $AB$  與向量  $AC$  的合，也就是最後決定的目標位置。

#### 3.2. 力場的調整

在[6]中，虛擬力場係根據使用者所在的環境及操控的歷史決定。此動態計算虛擬力場的計算公式如下：

$$\vec{F}'_p = \mu_{user} \times \mu_v \times \mu_d \times \vec{F}_p \quad (1)$$

$\vec{F}_p$  是從預先建立的力場中，根據目前位置，取出的力場向量(如圖三的  $\vec{AC}$ )， $\vec{F}'_p$  則是經過調整完後的力場向量(如圖三的  $\vec{AD}$ )，也就是用來修正使用者操作向量(如圖三的  $\vec{AB}$ )的向量。 $\mu_v$  和使用者目前速度有關， $\mu_v$  定義為  $\mu_v = kv$ ；其中  $v$  為使用者目前的速度， $k$  則是比例常數。 $\mu_d$  則和方向有關， $\mu_d$  定義為  $\mu_d(step_n) = \eta \times \mu_d(step_{n-1})$ ，其中  $\eta$  值的決定主要是看目前前進方向的向量和  $\vec{F}_p$  的內積，大於零時， $\eta = m$  ( $m > 1$ ，為一個放大常數)；小於零時， $\eta = n$  ( $n < 1$ ，為一個縮小常數)。 $\mu_{user}$  和使用者喜好有關， $\mu_{user}$  的定義為  $\mu_{user} = l$ 。喜歡力場整體強度大一點的使用者， $l$  的值就需要設大一點，反之亦然。



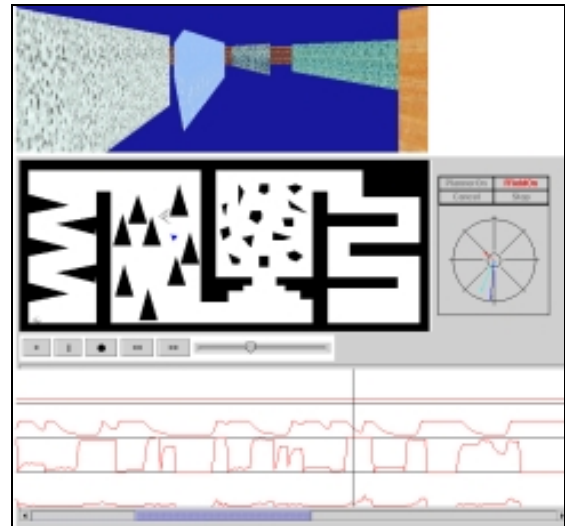
圖四：系統運作的架構圖

在[6]中， $(l, k, m, n)$  這四個參數（稱為一組參數組態）都是採用固定的比例常數。但是在實驗中也發現，這些參數的效果和場景的變換及使用者的特性都有關係；使用者會有自己偏好的值，沒有一個絕對的標準。即使要找到一組適合大部分使用者的參數，也只能靠不斷的實驗累積，從經驗中求得，但還是沒有辦法滿足全部的使用者。因此，我們希望能夠建立一個可以自動根據不同使用者特性對參數作最佳化的虛擬輔助力場。

#### 4. 可調適之輔助力場的設計

根據上述實驗的經驗，我們知道虛擬力場的輔助方式，必須因人而異的適性調整，才能得到最佳的效果。因此，我們設計了兩套不同的機制：「模擬實驗」及「動態調整」，分別為離線及線上兩種情形，實做出可依照使用者操控行為而自動調適虛擬力場的瀏覽輔助模組。圖四為此可調適輔助力場的系統架構圖。系統輸入來源可以是真實的使用者，或者是系統模擬產生的資料。傾向分析與修正模組是用來修正模擬時產生的輸入資料，以符合使用者的高階傾向。虛擬力場輔助模組則是負責將使用者輸入和虛擬力場合成運算後，調整目標點位置，以幫助使用者減少碰撞。力場參數產生器則是根據模擬的結果將一組力場參數組態送給虛擬力場輔助模組，以做為虛擬力計算的依據。

第一種可調適力場的設計方法是「模擬實驗 (simulation experiments)」。我們將使用者的瀏覽行為為取樣；透過取樣資料，在離線狀態下(off-line)，利用程式模擬使用者在所有可能參數組態下的瀏覽情況，並找出到達終點所走步數最少的那組最佳參數組態，以做為該位使用者下次進入此虛擬場景的力場參數組態。一次模擬實驗中我們只使用一組



圖五：實驗的圖形介面，最上面是 3D 顯示的部分；中間為平面場景對應圖以及互動過程錄製及播放功能；最下面是所受的力場力量的數值趨勢圖；右邊則是力場修正計算方式的示意圖。

固定的力場參數組態；而在一次模擬實驗過程裡，我們將取樣資料中的每一筆操控紀錄循序輸入 3D 介面中，系統會先作傾向分析與修正，再經過虛擬力場輔助後，才會改變下一個視點的座標並更新 3D 顯示畫面。在設定範圍內全部的力場參數組態都測試完之後，所有模擬結果中表現最好的組態即是該使用者在此場景下最佳的力場參數組態。我們將在第 5 節中進一步介紹模擬實驗的詳細作法。

第二種方法稱為「動態調整(dynamic adjustment)」。使用者不需要先取樣後再離線分析，系統會在瀏覽的過程中根據使用者的反應直接調整力場參數組態。這個方法在實做上是採用類似模擬實驗的方法，不過取樣和模擬的範圍縮小，以達到即時的模擬及調整。輔助的虛擬力場會持續的根據使用者在虛擬環境中的操控行為，即時擷取一小段歷史資料，並且在互動介面一個迴圈的時限內，對這些片段取樣的資料作模擬實驗，找出最佳的參數組態，並透過力場參數組態產生器，將此組態在下一個迴圈提供給虛擬力場模組使用。在第 6 節裡，我們會對動態調整的方式，作進一步的說明。

#### 5. 使用模擬實驗找出最佳力場參數組態

##### 5.1. 瀏覽過程之取樣

從以虛擬力場輔助來改善瀏覽介面的研究報告裡，我們發現在虛擬力場的輔助下，使用者從出發點走到目的地所花費的時間的確減少很多。但是如果想要做到可以針對不同的使用者調整，我們必須先把輔助的機制和使用者的反應量化，以分析



瞭解輔助機制有效的原因。因此，我們設計了一個使用者操控過程錄製與播放的機制（如圖五），將使用者瀏覽的過程記錄下來，以做為模擬實驗的取樣資料來源。除此之外，由於此記錄機制能提供類似錄影機的撥放、進、退等的功能，因此可以做為日後方便實驗觀察及分析之用。

在一個場景的取樣過程中，我們讓使用者從設定的起點走到設定的終點以完成一次瀏覽任務。在瀏覽過程中，我們記錄的資料包含行走的速度（滑鼠拉出的垂直分量）、旋轉角度（滑鼠拉出的水平分量）、及整段行走的軌跡等。我們將這些資料以檔案形式儲存下來，以做為將來在模擬實驗過程中，人機介面輸入資料的來源。

## 5.2. 模擬實驗

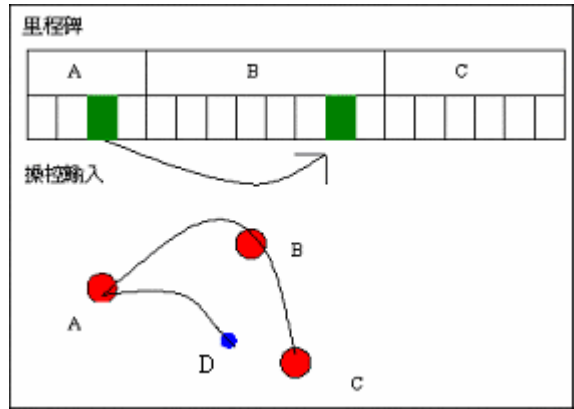
在瀏覽過程的每一刻，力場大小的決定，是由 3.2 節提到的四個參數 ( $l, k, m, n$ )，以及使用者當時的速度和方向才能決定；這使得探討使用者在不同輔助程度下的差別困難了許多。我們希望讓系統可以根據每個使用者的特性找尋適合他們的最佳參數，但又不希望讓每個使用者親自測試過所有的參數組態，因為這樣不但沒效率而且不切實際。因此，我們以取樣和自動模擬的方式讓最佳參數組態的搜尋過程自動化。換言之，針對個別的使用者，我們使用 5.1 節的取樣機制得到使用者的取樣資料，再將取樣資料送回程式中，模擬出使用者在各種不同力場參數組態下的表現，並找出最佳的一組。這裡所謂最佳參數組態的定義，就是能輔助使用者在最少步數內由起始點抵達指定終點的參數組態。

### 5.3. 使用傾向分析對修正模擬輸入

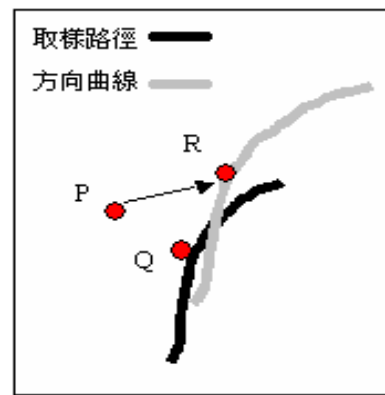
為了正確模擬使用者的行為，我們在模擬實驗過程中加入了使用者的傾向分析，以修正模擬實驗的輸入資料。一般使用者在受到瀏覽輔助時，會根據視覺回饋，修正輸入命令（滑鼠拖曳長度）的大小。但是在電腦模擬的過程中，由於使用不同的虛擬力場參數，因此視點的運動軌跡將與原取樣時不同。但由於電腦模擬缺乏視覺回饋的機制，因此我們必須能找出使用者在瀏覽路徑過程中每一點的原始傾向，才能根據此原始傾向進行必要的修正，以減少實驗模擬的誤差。本研究從取樣的資料中，分析出「里程碑」和「方向曲線」兩項指標，以做為使用者的高階傾向，修正模擬使用者的低階輸入資料。

#### 5.3.1. 里程碑

我們在使用者取樣時所走出來的路徑上，每隔一定的距離放上一個里程碑，如圖六所示。從起點到終點一路放下去，利用里程碑將取樣資料中的操控輸入分段。在模擬實驗進行時，根據目前所在



圖六：ABC 三點為里程碑，D 點是目前的位置，因為離目前點最近的里程碑是 C，所以略過一些多餘的操控輸入。



圖七：利用方向取線修正模擬中可能偏差的方向

的里程碑，找出當初取樣過程中，相對應的操控輸入。因為我們在輔助的情況下，或許可能會從 A 里程碑就直接走到 C 里程碑，這時候在 B 里程碑的那段操控輸入就變成不需要了，或是原本從 B 和 C 里程碑中，需要耗費很多時間才走過去，相對的也會有很多的操控輸入。但是有了輔助功能後，花費比較少的操控輸入很順利的就從 B 走到 C 里程碑了，這時我們就捨棄掉那些多餘的操控輸入。

#### 5.3.2. 方向曲線

方向曲線主要是修正模擬時方向偏掉的問題。我們假設人在行走時，都有長程的目標和短程的目標；長程就是終點，至於短程我們認為是眼前一小段距離的地方，因為人前進的方向應與正前方視線方向相差不多。我們把取樣資料中每一步的這些點連起來，就成了前進時的方向曲線。如圖七所示，在模擬過程中，先根據目前的位置 P，從取樣資料的路徑裡找到離目前最近的一點 Q（可以利用里程碑縮短搜尋的速度），再找出 Q 點對應到方向曲線上的點 R，PR 連線就是我們修正後的方向。這種做法的涵義，其實是讓模擬過程中，每走一步

都是有一個目標在引導著他走向終點，而引導的路線就是之前使用者所走每一步的近程目標。這樣一來可以避免只根據紀錄下來的資料行走，而沒有考慮到目前環境的變化和輔助力場改變的問題。

## 6. 動態調整力場參數

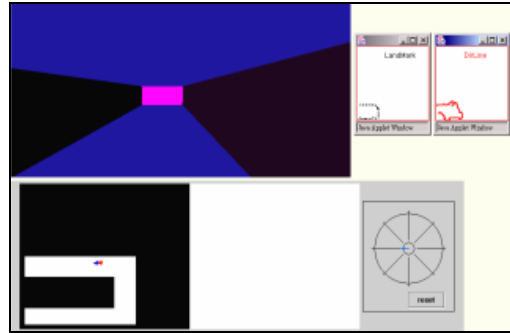
用模擬實驗的方法，最大的缺點就是需要花費較多的時間在模擬的過程，而且所找到的最佳參數組態可能不見得適用於其他場景。目前輔助力場的設計裡，有四個需要調整的參數。假設每個參數的值在許可範圍內都可以有  $K$  個變化，這樣總共需要  $K^4$  次模擬； $K$  越大表示模擬的程度越細，找出來的參數也會越趨近對使用者最有利的參數，但相對而言時間花費也越多。目前考慮效率問題我們將  $K$  定為 3，也就是總共會進行 81 次模擬實驗，以決定最佳組態。不過即使將  $K$  定在 3，使用者還是需要等待一段分析時間，才有辦法找到最佳的參數組態。為了改善這個問題，我們以模擬實驗的精神，設計出一個可以線上方式動態調整虛擬力場參數的機制。

### 6.1. 縮小取樣及模擬實驗範圍

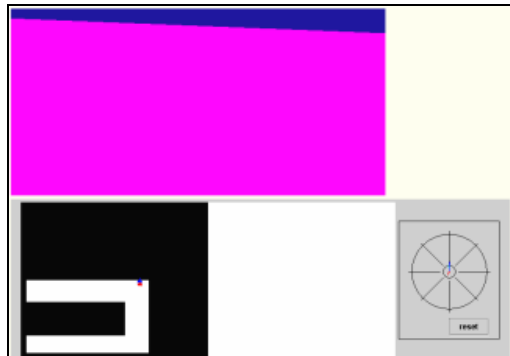
在此動態調整的方法裡，我們在模擬實驗加入了一個大小為  $N$  的滑動視窗；此滑動視窗內，我們會一直保持使用者最近  $N$  步的滑鼠輸入資料。假設目前使用者已經走了  $K$  步，滑動視窗中的資料就是第  $K$  步到  $\max(K-N+1, 0)$  步的滑鼠取樣資料；同時，我們也紀錄每一步所走到的座標位置。當使用者瀏覽超過  $N$  步之後，每多走一步系統便會以模擬實驗搜尋出適合前  $N$  步的較佳參數組態，並以此參數組態做為下一步移動時虛擬力場的輔助方式。在此動態調整的方法裡，除了模擬的長度不同外，參數組態的搜尋範圍亦不相同。假設由四個參數所構成的組態為  $(A_1, A_2, A_3, A_4)$ ， $\Delta_n$  為參數  $A_n$  一次調整的單位量，則此組態的相鄰組態為  $(A_1-\Delta_1, A_2, A_3, A_4)$ 、 $(A_1+\Delta_1, A_2, A_3, A_4)$ 、 $(A_1, A_2-\Delta_2, A_3, A_4)$ 、 $(A_1, A_2, A_3, A_4+\Delta_4)$ 。換言之，我們把模擬的範圍縮小為組態本身和相鄰的所有組態。我們評估的標準是走的距離，越長代表效率越好。因此，從這九個組態模擬的結果中，我們找出行走距離最遠的一組，並把目前的力場參數換成該組。如此一來，力場參數將逐步在使用者的行進過程中調整，趨近於該使用者最合適的參數。

### 6.2. 將參數惰性化以避免變化過於頻繁

我們發現使用動態調整力場參數組態時，當走的步數超過  $N$  步以後，每多走一步力場參數就可能因視點與場景相對關係的變化而調整一次，如此一來，最佳力場參數的變化將有過於頻繁的情況發生，使用者也可能因此而覺得操作介面不自然。為了解決這個問題，我們將參數套上惰性。我們不



圖八：利用里程碑與方向取線做修正的實驗模擬，順利的通過了轉彎的場景。



圖九：未做修正的實驗模擬，轉完彎道後，卻卡在牆邊。

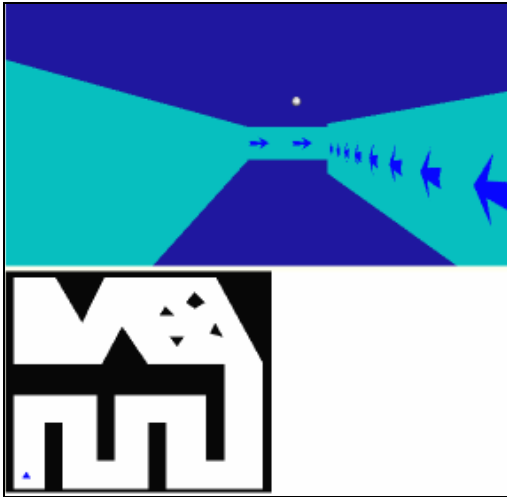
直接把模擬出來的最佳參數組態直接於下一步使用，而是每個組態都會替變化的特定參數累積或減少能量（參數預設基本能量都為 0）。例如假設模擬結果最好的組態為  $(A_1+\Delta_1, A_2, A_3, A_4)$ ，則  $A_1$  這個參數將累積一個能量； $(A_1, A_2-\Delta_2, A_3, A_4)$  則是  $A_2$  將減少一個能量。當參數  $A_n$  累積或減少到 3 個能量時，我們才對參數  $A_n$  增加或減少一個單位的量。這樣可以避免參數跳動過於頻繁，也可以確保參數  $A_n$  的確是有上升或下降的趨勢時，才去調整它。

## 7. 實驗結果及討論

本論文所使用的系統為自行以 JAVA 語言開發的虛擬力場調整機制。在圖形化介面中的 3D 呈現部分（如圖五所示），則是使用 Blaxxun 公司[1]所出的 Blaxxun 3D VRML 瀏覽器。

### 7.1. 使用與未使用傾向分析比較

為了比較使用傾向分析修正和未使用傾向分析修正的模擬實驗，我們建立了一個簡單的場景，如圖八。在 3 次的取樣過程中，如果沒有使用傾向分析修正，在使用預設的力場參數下，3 次都無法順利的從起點走到終點。若使用了傾向修正，3 次模擬過程都可以順利完成。圖八和圖九分別是



圖十：實驗場景，場景中所標示的箭頭是為了指引使用者瀏覽方向。畫面中白色的球，則是終點上方。使用者則是循著箭頭，瀏覽整個場景走到白球所在地。

使用和未使用傾向分析的實驗過程，我們可以看到在圖八中，模擬的過程順利的通過轉彎的地方；在圖九中，則是卡在轉角過去不遠的地方。

## 7.2. 模擬實驗與動態調整的比較

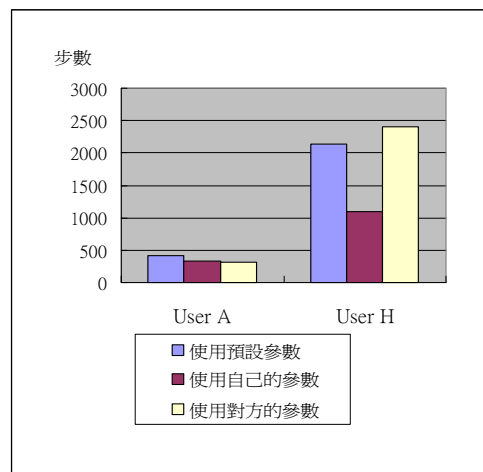
在模擬實驗和動態調整的實驗裡，我們找了十位使用者。這十位使用者中，有兩位對 VRML 虛擬環境操作熟悉(A、B)，兩位對 3D 遊戲操控非常熟悉(C、D)，有三位則是對電腦操作熟悉(E、F、G)，但是不常接觸 3D 及 VRML 虛擬環境操作。剩下三位使用者中，兩位很少使用電腦，一個月約 1-2 次(H、J)，一位則是從來不用電腦(I)。實驗場景如圖十，我們對每個使用者分別做模擬實驗和動態模擬兩部分的測試。

在模擬實驗的部分，我們先讓使用者在沒有力場輔助的情況下取樣，再進行模擬實驗的分析。分析完後，我們請使用者分別使用預設力場參數和模擬實驗分析找出來的最佳參數，各進行兩次瀏覽，並且紀錄所走的步數和時間，最後取兩次的平均。實驗紀錄為表一的無力場（取樣過程）、預設力場參數及模擬實驗三欄。其中有兩個使用者 C、D 在調適後反而走的比預設的參數組態差，不過整體平均比預設組態好將近 8%。使用者 I (不曾使用電腦) 甚至從預設組態下的 2143 步(110 秒)下降到 1096 步(54 秒)。

在動態調整的部分，我們任意取了三組不同的起始參數組態。我們讓每一個使用者，對每一組參數組態分別作三次瀏覽，最後再將三個組態所測得的九次瀏覽所花費的步數和時間求平均。實驗紀錄為表一的動態調整一欄。有四位使用者 A、C、D、H 使用動態調整結果和模擬實驗的結果，所走

表一：四種情況下抵達終點花費步數與比例(單位：步)。預設力場、模擬實驗和動態調整三欄的百分比是和無力場輔助模式下比較的結果。

User	無力場		預設力場參數		模擬實驗		動態調整	
	Step	%	step	%	step	%	step	%
A	779	100	407	52.2	328	42.1	325.6	41.8
B	902	100	416	46.1	320	35.5	424.7	47.1
C	1838	100	491	26.7	548	29.8	555.7	30.2
D	1199	100	458	38.2	607	50.6	614.1	51.2
E	834	100	510	61.2	406	48.7	439.9	52.7
F	1715	100	840	49.0	792	46.2	585.8	34.2
G	1338	100	950	71.0	716	53.5	735.4	55.0
H	4111	100	987	24.0	771	18.8	803.9	19.6
I	7129	100	2143	30.1	1096	15.4	2150.4	30.2
J	1927	100	1061	55.1	725	37.6	963.4	50.0
Avg	2177.2	100	826.3	45.4	630.9	37.8	759.9	41.2



圖十一：使用者 A 和 H 對彼此的最佳化參數組態反應相差很多。A 影響不大，H 甚至比使用預設的力場參數組態還差。

的步數相差在 1% 之內，使用者 F 甚至只要走 586 步(37.7 秒)比使用模擬實驗所需 792 步(54.3 秒)快了 8%。

我們從上個實驗的十位使用者中，挑出兩位，使用者 A 和 H，來進行使用者個別差異的比較。我們分別讓兩位使用對方在模擬實驗中找出的最佳力場參數組態，也就是讓 A 使用 H 的最佳力場參數組態，讓 H 使用 A 的最佳力場參數組態，然後讓兩人進行兩次瀏覽實驗(固定力場參數組態，沒有使用動態調整)。圖十一為兩次實驗平均所花費的步數比較。使用者 A 和 H 對彼此的最佳化參數組態反應相差很多。使用者 A 的表現和使

用自己的最佳化參數差不多，但使用者 H 在使用了 A 的參數組態後，甚至比使用預設參數組態還差。這說明了使用者個別差異確實存在。

### 7.3. 討論

由於使用者差異確實存在，只使用一組預設的虛擬力場參數組態勢必無法滿足全部的使用者，因此需要針對不同的使用者去做個別調適。從表一可以看出，兩種方法平均比較起來，不論是就所走的步數或花費時間，使用動態調整的方法要比單純使用預設力場參數好，而使用模擬實驗的方法又比使用動態調整好。但是使用模擬實驗需要花費比較多的分析時間。而使用動態調整的方法所得到的效率雖然平均還是沒有模擬實驗來的好，但是從個案分析中可以得知效率其實和模擬實驗相差不遠，而且動態調整不需要額外的分析時間。就算隨意給一組起始參數組態，動態調整的方式也可以慢慢調整到適合使用者的參數組態。

### 8. 結論與未來研究

由環境中的障礙物所產生的力場的確可以幫助減少碰撞，使得瀏覽更順暢，但是此虛擬力場的使用方式具有個別差異。為了改善這個問題，我們採用了模擬實驗和動態調整兩種方式來進行個別差異下力場輔助機制的探討。模擬實驗使用取樣和模擬的方式評估使用者在不同參數組合下的瀏覽效率，找出效率最佳的參數組態。動態調整方式在方法上類似模擬實驗的精神，但在實做上則是縮小取樣和模擬的範圍，利用漸進的方式調整使用者的參數組態。從實驗中發現，使用模擬實驗方式提升的瀏覽效率最好，但是需要花費較多時間在離線狀態下(off-line)做模擬分析。而線上(on-line)動態調整的方法不需要花額外的分析時間，而且提升的效能也只會比模擬實驗差一點而已。另外，在模擬實驗中我們還採用了里程碑和方向曲線這兩項指標來做為使用者高階意向的依據，以修正模擬實驗的低階輸入。在動態調整參數組態的方法中，我們替參數加上了惰性，以避免參數組態變動太頻繁，造成使用者不自然的感覺。我們發覺，這些機制讓我們的模擬實驗更加精確，而實驗所得的最佳參數組態，也都能反映個人化的差異，提高虛擬環境中 3D 瀏覽的效率。

目前為了簡化系統架構，暫時沒有考慮運動計劃器輔助的部分，希望未來能將運動計劃器的調適一起整合進去。另外，目前研究的假設是先不考慮場景的影響。在未來加入場景的考量時，將會進一步考量心理學方面的因素，例如視野的寬闊、年齡性別、路標顯示方式等如何影響人類的行走模式。這些都是這個研究未來延伸的目標。

### 致謝

此研究在國科會 NSC 91-2213-E-004-005 及 NSC 91-2815-C-004-001-E 計畫的支助下完成，特此致謝。

### 參考文獻

- [1] Blaxxun3D. An embedding java applet VRML browser. Information is available at <http://www.blaxxun.com>.
- [2] J. Edwards and C. Hand, "Maps: Movement and planning support for navigation in an immersive vrml browser," in *Proc. of VRML'97 Conference*, 1997.
- [3] P. K. Egbert, and S. H. Winkler, "Collision-Free Object Movement Using Vector Fields," in *IEEE Computer Graphics and Applications*, 16(4):18-24, July, 1996.
- [4] J.-C. Latombe, *Robot Motion Planning*, Kluwer Academic Publisher, Boston, MA, 1991.
- [5] B. Lenzmann, I. Wachsmuth, "A User Adaptive Interface Agency for Interaction with a Virtual Environment," in *Working Notes of the IJCAI-95 Workshop on Adaptation and Learning in Multi-agent Systems*, 43-46. AAAI Press, 1995.
- [6] T.Y. Li, H.C. Chou, "Improving Navigation Efficiency with Artificial Force Field," in *Proc. of 2001 14th IPPR Conference on Computer Vision, Graphics, and Image Processing*, Taiwan, 2001.
- [7] T.Y. Li, and H.K. Ting, "An Intelligent User Interface with Motion Planning for 3D Navigation," in *Proc. of the IEEE Virtual Reality 2000 Conference*, P177-184, New Jersey, 2000.
- [8] Desney S. Tan, George G. Robertson, Mary Czerwinski, "Exploring 3D Navigation: Combining Speed-coupled Flying with Orbiting," in *Proc. of the SIGCHI Conference on Human Factors in Computing Systems*, March 2001.
- [9] Eric A. Wernert, and Andrew J. Hanson, "A Framework for Assisted Exploration with Collaboration," in *Proc. of IEEE Visualization'99*, San Francisco, October 1999.
- [10] D. Xiao, R. Hubbard, "Navigation Guided by Artificial Force Fields," in *Proc. of the ACM CHI'98 Conference*, pp179-186, 1998.
- [11] L. Hong, S. Muraki, A. Kaufman, D. Bartz, and T. He, "Virtual Voyage: Interactive Navigation in the Human Colon," in *Proc. of SIGGRAPH'97 Conference*, pp27-35, 1997.
- [12] Neilson and Olsen, "Direct Manipulation Techniques for 3D Objects Using 2D Locator Devices," in *Proc. of the 1986 Workshop on Interactive 3D Graphics*, pp175-182, 1986.