

## 第三章

### 系統總覽

在介紹了運動計畫及虛擬人物動畫模擬的相關研究之後，本章接下來將介紹本論文的系統總覽，3.1 節說明系統的架構與流程；3.2 節介紹環境初始的過程；3.3 節簡介系統的運動計畫器；最後的 3.4 節介紹動畫生成的元件。

#### 3.1 系統流程與架構

根據前面章節所述，本論文的目的是在一個佈滿梅花樁的虛擬環境下，在給定舞獅人物運動起點和終點之後，藉由我們所設計的運動計畫器，規劃出一條舞獅人物可行走的路徑。本系統的架構大致可分為三大部分：第一部份為環境初始元件，目的在於產生佈滿梅花樁的環境，以及供使用者設定初始值；第二部分為運動計畫器，主要功能是在上述環境搜尋出一條可供舞獅演員行走的路徑；第三部分為動畫產生元件，功能包括舞獅人物動作的調整以及動畫輸出。

圖 3.1 為本論文所提之系統流程圖，環境初始首先需要建立數十個高度不同且分佈區域不一致的梅花樁，然後由使用者設定舞獅演員運動起始點及終止點。本系統的運動計畫器是採用最佳化搜尋演算法的精神，依據舞獅演員運動的習性和規則來建立而成，運動計畫完的輸出結果是一連串的脚步座標值。在動畫產生前還有一個前置步

驟是估算相對於每個步伐人物重心的位置，連同上一步所產生的腳步座標值一併輸出至動畫產生系統。而我們動畫產生的機制是透過 RhyCAP 系統以程序式的方式自動產生動畫。

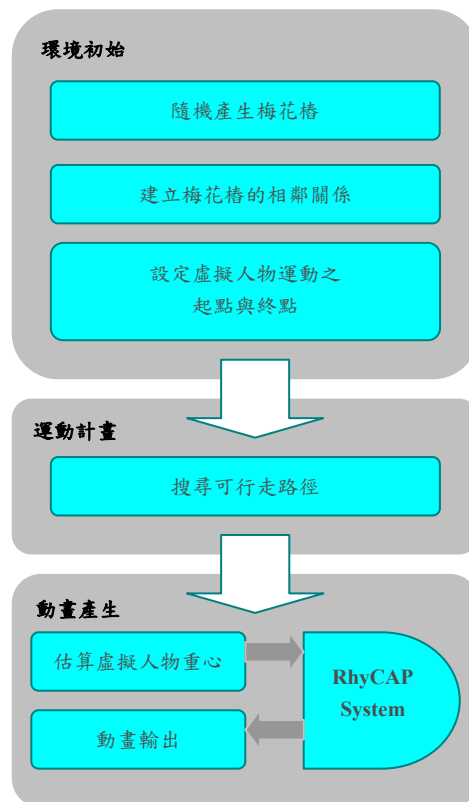


圖 3.1：系統流程圖

### 3.2 環境初始

真實世界中舞獅演員所跳的梅花樁一般都是由兩行的高低樁組成，並呈現一定型態的高低起伏，以符合舞獅演員的表演模式。而在我們的虛擬環境中，為了增加問題的挑戰性，我們預設的梅花樁為  $m*n$  的陣列， $m$  為梅花樁的行數， $n$  為梅花樁的列數。為了簡化場景生成的複雜度，梅花樁中固定不變的是樁與樁之間前後左右的距離，可變動的為每一列中的梅花樁數目和梅花樁的高度。描述一個梅花樁的參數包含  $X$ - $Y$ - $Z$  座標以及稍後介紹的鄰居陣列  $N$ 。第  $i$  個樁的參數可以表示為： $S_i(X_i, Y_i, Z_i, N_i)$

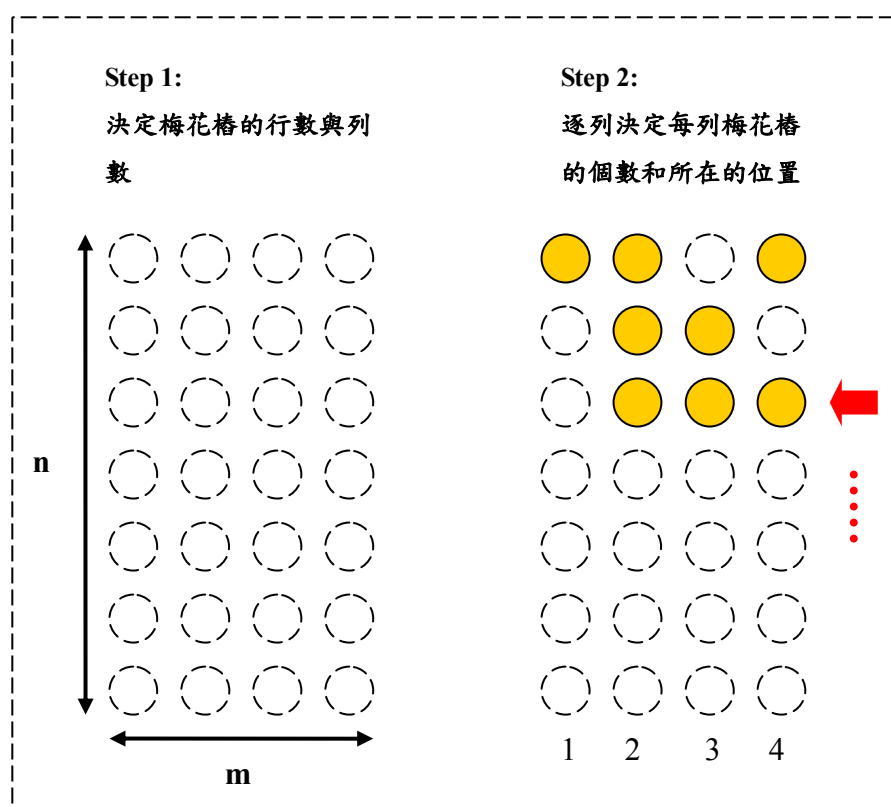


圖 3.2：梅花樁建立過程示意圖

梅花樁的產生是分三步逐列進行，第一步決定梅花樁的行數與列數。第二步決定每列包含梅花樁的個數以及每個樁在該列的位置。如圖 3.2 的示意圖，實心圓形為系

統產生的梅花樁。最後一步則決定梅花樁的高度，為了讓舞獅人物可以行走於列與列之間，因此每列的梅花樁有一個基準高度，相鄰兩列梅花樁中基準高度不會相差太多，這樣產生出來梅花樁的坡度才不致變化太大。每個梅花樁的實際高度是以所在那一列的基準高度，上下作小範圍的增減；而每列梅花樁的基準高度是根據前一列梅花樁的基準高度作一定範圍的增減。根據以上規則我們可以依序產生梅花樁，如圖 3.3 場景中的梅花樁即為系統所產生的測視圖及上視圖。

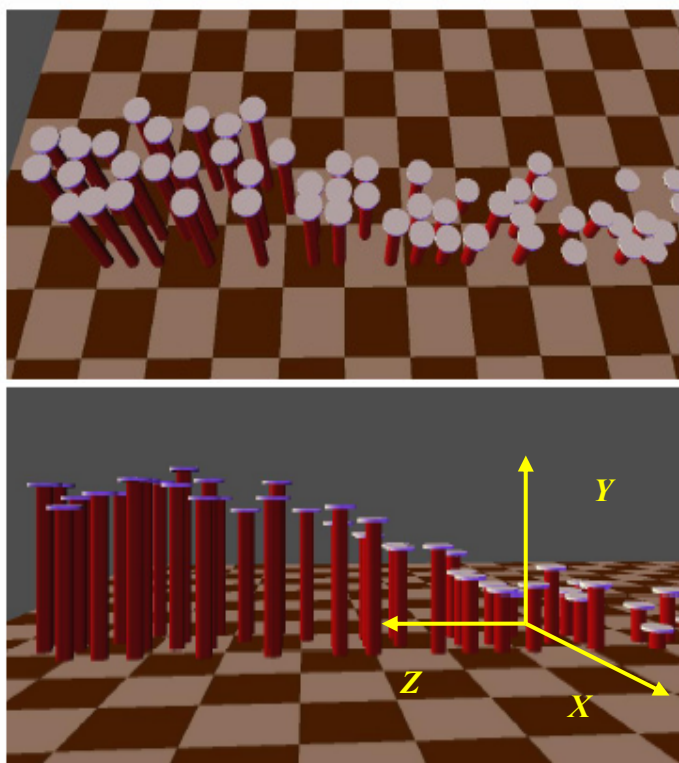


圖 3.3：梅花樁上視圖與測視圖

梅花樁產生後，接著系統將建立梅花樁之間的相鄰關係。此關係的主要概念是針對每個梅花樁以自己為中心，用一固定半徑畫圓形成一個圓形區域，落在這個區域內的其他梅花樁都定義為中心梅花樁的鄰居。從虛擬人物的角度來看，也就是說當一隻腳落在某個梅花樁時，相鄰的梅花樁為另一隻腳可能可以踩到的範圍。另一方面，舞獅演員的運動型態可分為走和跳躍兩類，因此我們會對每個梅花樁定義兩類的鄰居，

近程的為可以由行走或跳躍運動到達，如圖 3.4 紅色圓形區域中的梅花樁；遠程的只能由跳躍運動到達，如圖 3.4 藍色圓形區域中的梅花樁。至於舞獅演員每一步伐選擇跳躍或行走的規則，在之後章節會再詳細介紹。梅花樁鄰居關係的建立之後，由於原來需要拜訪除了自己之外的每個樁，變成只需拜訪自己的鄰居，所以在運動計畫的過程中搜尋的空間及時間將大幅的減少。舞獅人物運動始點跟終點，可以由使用者設定，如果使用者沒有特別指定，預設便是由梅花樁陣列的一端做為起點出發，另一端為終點。

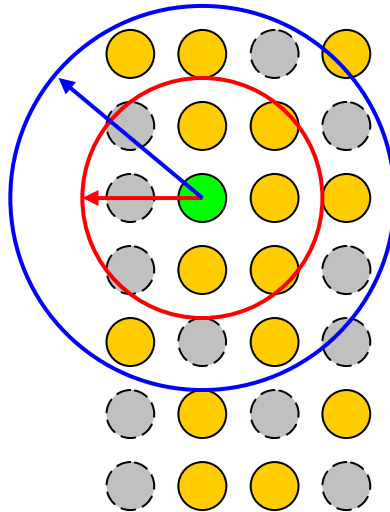


圖 3.4：梅花樁鄰居關係建立示意圖

### 3.3 運動計畫器

虛擬環境中的梅花樁建立完成之後，梅花樁的座標和高度資訊會送到系統中的運動計畫器。運動計畫器的功用便是在舞獅人物運動的起點和終點之間，找尋一條舞獅

人物可以順利通過的路徑。這當中必須考慮人物運動的限制，例如太遠的樁用走的方式可能過不去，就得判斷是否可以用跳躍的方式完成。

運動計畫器採用的演算法根據最佳化搜尋演算法(Best First Search Algorithm)的概念來設計，在搜尋的程序中規劃出所有的步伐。計畫器的輸入參數為梅花樁的地形資訊和舞獅演員起點及終點的座標位置，規劃完成後則輸出一序列的人物步伐資訊，更進一步的運動計畫細節會在下一章詳細介紹。

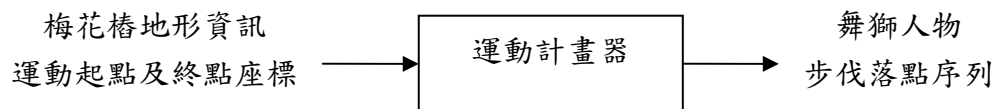


圖 3.5：運動計畫器示意圖

### 3.4 動畫生成元件

舞獅人物運動路徑規劃完成後，所產生的步伐資訊便會送到動畫生成元件，產生運動過程的各個關鍵格，再由關鍵格內插器完成整段動畫。本研究中所使用的虛擬人物模型，是沿用 RhyCAP 系統[2]中人物角色的模型，其中每一個演員是以樹狀結構的形式連結關鍵性的節點來表示，如圖 3.6 所示。此人物模型總共 15 個關節，總計有 40 個自由度，其中 12 個分佈在軀幹和頭部，剩餘的 28 個分佈在四肢。依據舞獅動畫的特性，我們把人物模型區分成兩大部分：軀幹和四肢，藉由 RhyCAP 系統中的 IK(Inverse Kinematic) Solver，要完成一個動作基本上只需指定四肢的終點(end-points or end-effectors)，IK Solver 會自動計算出各 IK Chain 中各關節點的角度和位移。

虛擬人物動作的產生，是使用 RhyCAP 系統的动作產生機制。基本上要模擬一個連續的動作，第一步要先建立適當的關鍵姿勢(key poses)，而描述一個關鍵姿勢包含指定虛擬人物雙腳的終點座標值和人物的重心座標值。例如，要模擬往前跨一步的動作，關鍵姿勢便是抬起自由腳、人物重心往前移動和放下自由腳三個程序。設定好關鍵姿勢後，中間剩下的過程就可透過以貝茲曲線內插的方式完成。此外，RhyCAP 系統中有建立一個腳步動作轉換圖(stance action graph)[2]，裡面包含舞獅動作的關鍵姿勢，並定義每個基本動作是包含哪些關鍵姿勢，透過追蹤動作轉換圖的不同節點，便可輕易的完成想要模擬的動作。

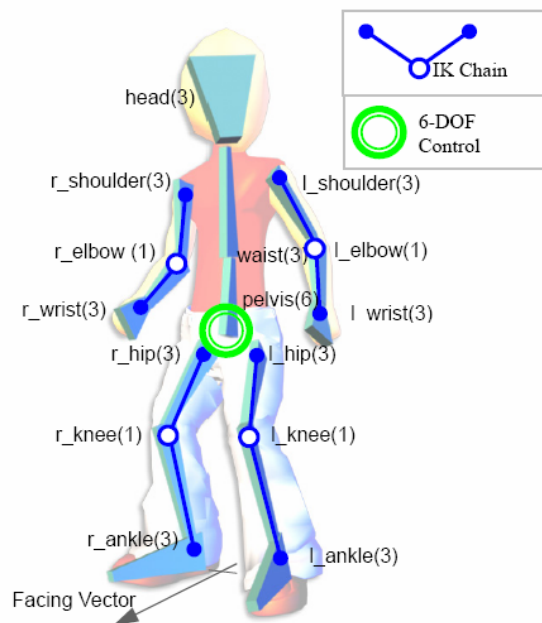


圖 3.6：虛擬人物模型