

以 Kinect 體感器實作之體感遊戲

– Move your Body

專題編號：100-CSIE-S006

執行期限：99 年 1 學期至 100 年 1 學期

指導教授：張厥煒

專題參與人員：學號：97590339 姓名：郭奕成

學號：97590342 姓名：黃臣逸

一、本文摘要

本專題發展一個新的人機介面互動方式，並以第三人稱冒險類的 3D 遊戲實作。

透過新的體感器—Kinect 裝置作為輸入端，取得人體骨架並對體感動作定義跑、跳、左右移動、前推、手勢滑動等姿勢，並透過圖學的處理把人體骨架轉換至遊戲模型上與遊戲整合，控制遊戲進行；遊戲方面也不只是單純實作一個遊戲，考慮到了遊戲架構的設計，使遊戲在畫面切換，流程控管中有序的進行。

本遊戲由於透過身體的動作來操作，增加了趣味性與真實性，對人機的互動更友善。

關鍵詞：人機互動、Kinect、體感動作定義、3D 遊戲、人體骨架、Postprocessing

二、緣由與目的

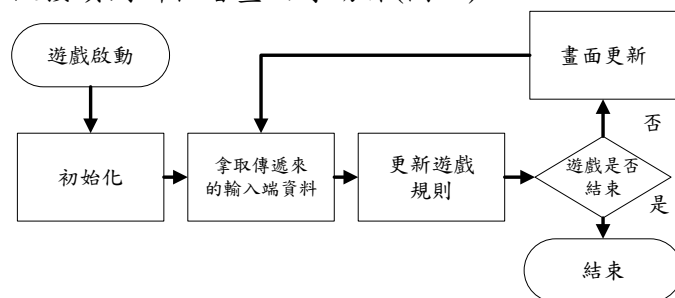
近來人機互動的觀念已逐漸重視，而今年「Kinect」體感器的誕生有戲劇性的變化，具備了攝影機所沒有的深度資訊，使辨識的難度降低，且價格普及，開發的成本亦減少，加速了互動的發展，成為未來的新媒介，產生了更多的應用可能與新的想法。

我們打算使用此裝置來擴展互動應用的可能性，但回顧以往眾多的互動模式當中，遊戲才是感受性最快的呈現平台，因此最後決定實作以 Kinect 做為輸入端的 3D 體感遊戲，帶給使用者人機互動所帶來的趣味性及真實的感受。

三、系統架構與流程

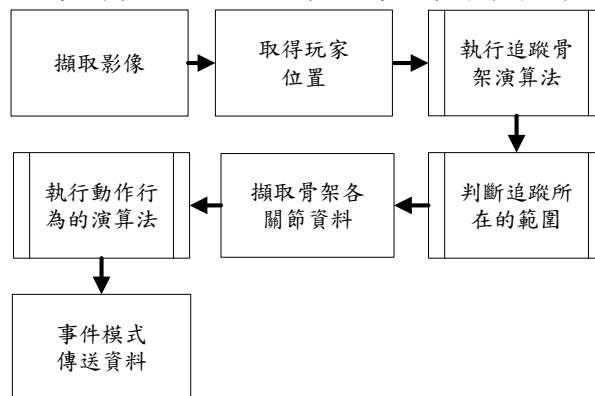
本遊戲分成遊戲端與輸入端；遊戲初始化並開執行緒啟動 Kinect 輸入端後，則輸入端會開始進行一連流程辨識動作並事件傳遞

給遊戲保存，遊戲更新資料時便會取出，並做接續判斷和繪畫面等動作(圖一)。



圖一：遊戲的主要流程圖

輸入端透過 Kinect 體感器取得影像，並拿取骨架資料後，實作體感動作演算，做為遊戲的輸入訊號，進行遊戲的操作(圖二)。



圖二：輸入端的處理流程圖

四、使用技術

(一) 使用工具：

1. **XNA4.0**：微軟提供的開發遊戲環境。
2. **JigLibX**：Open Source 的 3D 物理引擎。
3. **OpenNI**：和 Kinect 溝通的框架，並提供一套完整的 API。

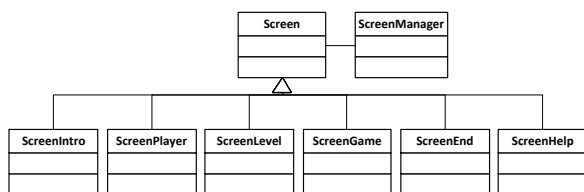
(二) 畫面與特效：

本專題使用三維空間來實作我們的遊戲，因此在圖學或 shader 特效的學習上花很

多工夫，共有：三維空間的 camera、3D 物理系統、3D 繪圖、2D 貼圖系統、天空盒子、水流、Postprocessing、3D 人物動畫管理與顯示、3D 圖層管理。

(三) 遊戲流程控管：

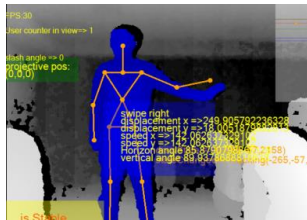
為了使遊戲流程順利，設計了一套遊戲管理的機制，藉由此機制可快速更改遊戲中參數與流程與畫面的切換設定（圖三）。



圖三：screen 管理的 UML 類別圖

(四) 影像擷取與人體骨架辨識：

Kinect 擁有深度辨識的功能，透過內部計算可獲得深度資訊與骨架節點（圖四）。



圖四：取得骨架-手勢滑動動作。

(五) 體感動作演算法：

取得骨架後，透過一些向量速度、角度構成的排列組合計算動作，若辨識成功則已事件發送訊息。

(六) 人物骨架與遊戲 3D 骨架模型結合：

取得骨架節點來源，透過基底矩陣轉換等方式轉換至遊戲人物的模型上，使其可以動作（圖五）。



圖五：實際動作對應遊戲中模型人物。

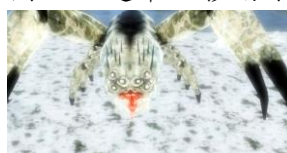
五、實作成果

遊戲取得輸入訊號後，即可操作選單（圖六）選擇關卡，關卡一是有隻猛獸在後頭追逐，玩家需要不斷跑且閃掉障礙到達目標處

（圖七、八、九）；關卡二是有牆壁不斷向前，牆壁會有挖洞，玩家須擺出某動作來閃避（圖十及圖五），各關卡結束後皆有計分來排名



圖六：選單，移動手勢到按鈕會顯示文字



圖七：遊戲中猛獸



圖八：抵達目標



圖九：關卡一畫面，遊戲中的障礙物及金幣



圖十：姿勢比對畫面

現在的
牆壁

通過數

七、結論

此體感器之出現加速了人機互動的發展，許多構想皆可能實現其應用。

本專題已實作出了許多符合人體動作行為，且透過與遊戲的結合也強調出互動與操作上的友善性，未來強化演算與架構設計也可套用至其他應用範圍如：互動藝術、互動式學習、醫療復健等領域，發展性明顯之多。

八、參考文獻

- [1]XNA4.0 動物書
- [2]XNA Sample Code
<http://create.msdn.com/en-US/education/catalog/>
- [3]OpenNI Google Groups
<http://groups.google.com/group/openni-dev>
- [4]OpenNI Document