

靜 宜 大 學

資訊工程學系

畢 業 專 題 成 果 報 告 書



資訊工程學系

學 生：

Computer Science & Information Engineering

資工四 A 411018426 蔡嘉哲

資工四 A 411017909 傅冠桀

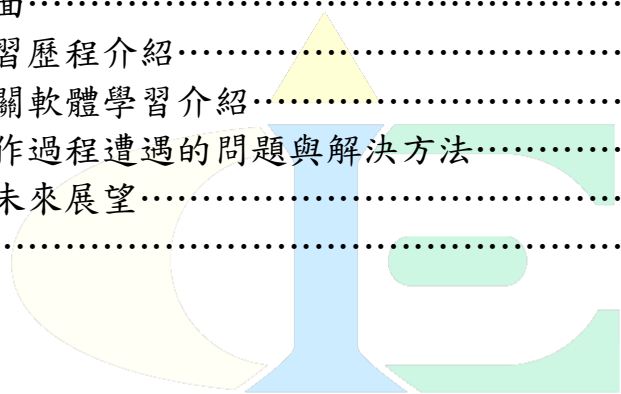
資工四 A 411017941 馬英傑

指導教授：劉國有教授

西 元 二 0 二 四 年 十 二 月

目錄

目錄	i
圖目錄	ii
第一章、	摘要.....	4
第二章、	研究與動機.....	4
第三章、	專題流程與架構.....	4
3.1	系統流程圖.....	4
3.2	系統架構圖.....	5
第四章、	專題成果介紹.....	8
4.1	軟體硬體設備資訊.....	8
4.2	系統畫面.....	9
第五章、	專題學習歷程介紹.....	12
5.1	專題相關軟體學習介紹.....	12
5.2	專題製作過程遭遇的問題與解決方法.....	12
第六章、	結論與未來展望.....	13
參考文獻	14

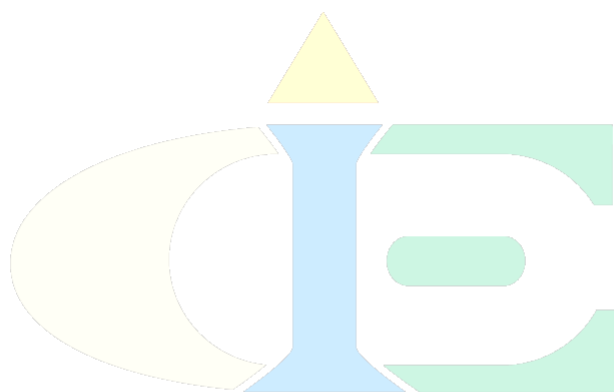


資訊工程學系

Computer Science & Information Engineering

圖目錄

圖 1	流程圖.....	5
圖 2	架構圖.....	8
圖 3	使用的軟、硬體.....	8
圖 4	開始畫面.....	9
圖 5	選擇介面.....	10
圖 6	mode 新手教學.....	10
圖 7	mode2 人機對戰.....	10
圖 8	mode2 結算畫面	11
圖 9	mode3 遊戲介面.....	11
圖 10	mode3 結算畫面.....	11



資訊工程學系

Computer Science & Information Engineering

第一章、 摘要

本專題利用 Unity 與 Kinect v2 開發的 AR 體感運動系統，與模擬拳手進行一場拳擊賽。同時具備新手教學與紓壓的打沙包模式，讓使用者在進行激烈的拳擊賽之餘，也有相對緩和的關卡來體驗。

運用 Kinect v2 的動作捕捉玩家動作，為使用者提供即時的互動回饋，增加運動的趣味性。透過系統的偵測，使用者可以在擊打的過程獲得反饋，提升 AR 運動的交互性。

第二章、 研究動機與目的

在現代社會，運動已經成為人們生活中不可或缺的一部分。而在眾多運動項目中，拳擊作為一項極具挑戰性和刺激性的運動，吸引著無數人的熱情和關注。然而，傳統的拳擊訓練往往需要大量的時間和精力，並且缺乏足夠的趣味性，對於一些初學者而言，可能會感到枯燥乏味。在這樣的背景下，擴增實境（AR）技術的應用為拳擊帶來了全新的可能性。

本專題將探討 AR 技術在拳擊中的應用，旨在通過 AR 體感運動，為體驗者提供一種全新的遊玩方式。透過 AR 技術，體驗者可以體驗在古代的競技場打拳擊賽，與對手進行對抗，可以即時獲得反饋，也同時確保了安全性；其次，在打沙包的場景中我們可以自行上傳圖片，想像在打擊討厭的人抒發壓力，給體驗者提供一些趣味性。

第三章、 專題流程與架構

3.1 專題流程

首先進入遊戲後可以看到開始畫面，包含進入遊戲以及退出遊戲，點擊進入遊戲後鏡頭會移到選擇介面，我們可以透過手的會揮動切換要進入的 MODE，雙手靠近頭部即可進入，當要退出時一樣雙手靠近頭部就可以退回選擇介面，點擊選擇介面右上角的按鈕即可退回開始畫面，點擊退出遊戲就可以離開程式。

具體流程如圖 1

負責在場景切換時保存和管理跨場景的數據（如主鏡頭位置）和狀態（如是否返回主場景）、當場景載入完成時，將主鏡頭移動到指定位置

a.5 Kinect_menu.cs

使用 Kinect 來檢測玩家的手勢，包含揮手和雙手抱頭動作。

手勢偵測有速度和時間限制，並且每個手勢有冷卻時間，以確保不會因重複偵測。一開始程式會延遲一段時間後再初始化 Kinect，以避免其他頁面或操作影響到程式的執行。

a.6 SwipeController.cs

菜單頁面的滑動和選擇功能，包括左右滑動和選擇當前頁面，並透過動畫效果移動頁面。

場景加載完成後，會恢復到上次保存的頁面，並根據當前頁面加載相應的場景。

a.7 CloseApplication.cs

退出遊戲

b. Model

b.1 PunchingBag.cs

根據碰撞檢測玩家擊打的不同部位，並通知 TutorialManager 更新擊打狀態，同時控制動畫和施加力的效果

b.2 ExitGame.cs

偵測雙手同時舉過頭部的動作來退出當前場景，並返回主選單

b.3 TutorialManager.cs

管理教學流程，根據玩家擊打的部位，逐步引導完成不同部位的擊打任務，並在教學完成後切換到自由模式

b.4 KinectManager.cs

Kinect 元件最基礎的部分。它控制感測器並對資料流進行處理，其他所有元件都依賴它提供的資料

b.5 AvatarControllerClassic.cs

將模型骨骼指定到 Kinect 追蹤的關節，我們選擇控制上半身

c. Mode2

c.1 Hit Box.cs

當玩家或對手擊中對方的頭部或身體時，根據條件計算得分，同時觸發對應的動畫和音效效果。以及通過事件暫時禁用頭部和身體的得分功能，以反映防禦和後退狀態

c.2 Model2.cs

控制敵人的行為，包括隨機出拳、防禦和後退，並根據玩家手勢偵測觸發對應的行動與動畫

c.3 Punch motion detection.cs

使用 Kinect 偵測人體關節位置分析手部和身體的移動來辨識揮拳、後退、防禦等動作，以及基於速度、移動距離和冷卻時間的條件，判斷動作並觸發對應的動畫和訊號事件（如揮拳或防禦）。

c.4 ScoringSystem.cs

管理玩家和對手的分數計算，同時實現倒數計時功能，控制遊戲時間與開始倒數，並根據遊戲時間或條件，顯示結束畫面，並暫停遊戲運行，等待玩家重新開始。

d. Mode3

d.1 Feedback.cs

處理遊戲中的反饋和計算連擊（combo）功能。當玩家進行打擊時，會註冊打擊數據並計算連擊次數，並根據打擊次數生成對應的反饋文字。

d.2 Punch.cs

利用 Kinect 進行揮拳偵測，當偵測到玩家揮動左右手時，會對物體施加力並觸發螢幕震動，並且記錄和反饋玩家的揮拳動作。

d.3 upimage.cs

在沙袋上顯示玩家上傳的圖片。

d.4 game3.cs

管理遊戲的倒數計時、計時和結束過程，並在遊戲期間監控玩家的打擊數據。當遊戲結束時，它會顯示統計資料並允許玩家重新開始遊戲。

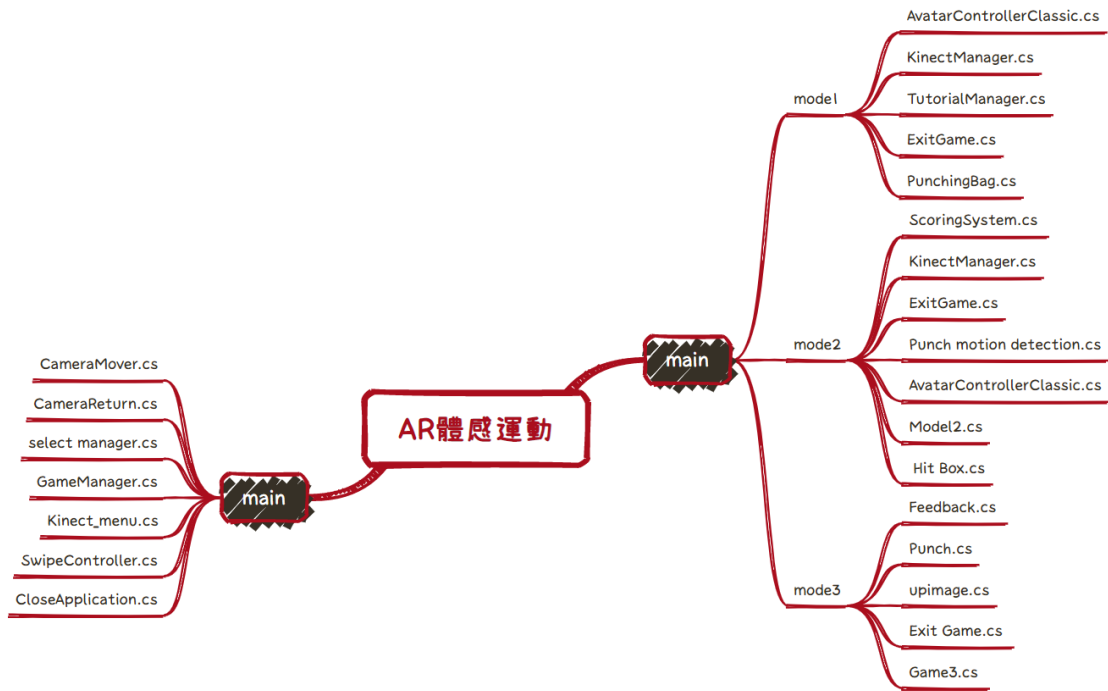


圖 2 架構圖

第四章、專題成果介紹

資訊工程學系

Computer Science & Information Engineering

4.1 軟體硬體設備資訊

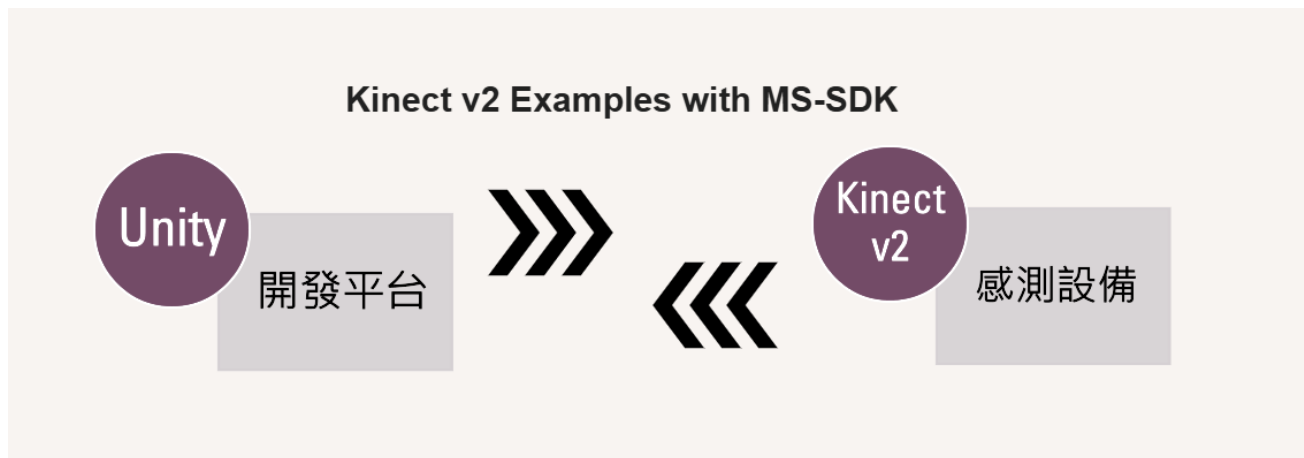


圖 3 使用的軟、硬體

Unity

Unity 是一個強大的跨平台遊戲引擎，廣泛用於開發 2D、3D 遊戲和互動式應用程式。它支援多種平台，包括 PC、手機、主機和 VR 裝置，並提供簡易的編輯器界面和強大的 API、直觀的界面和完整的工具集，適合快速開發和原型設計。Unity 使用 C# 作為主要編程語言，擁有龐大的資源庫和社群支援，是許多創作者實現互動內容的首選工具。

Kinect v2

微軟推出的一款體感裝置，以精準的深度感測和人體骨架追蹤功能聞名。它能透過內建的深度攝影機和紅外線技術，實現對人體姿勢、動作和手勢的即時追蹤，並提供聲音辨識功能。相比於初代，Kinect v2 在感測範圍、解析度和追蹤穩定性上都有明顯提升，適合用於遊戲控制、互動藝術裝置以及醫療和運動分析等領域。結合 Unity 使用時，Kinect v2 能大幅提升互動應用的沉浸感與趣味性。

4.2 系統畫面



圖 4 開始畫面

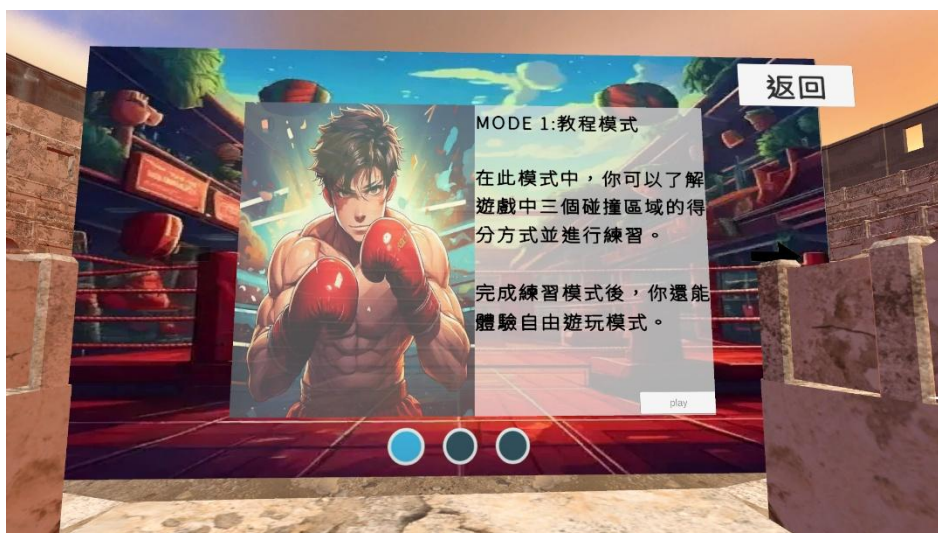


圖 5 選擇介面



Computer Science 圖6 model1 新手教學 Information Engineering



圖 7 mode2 人機對戰

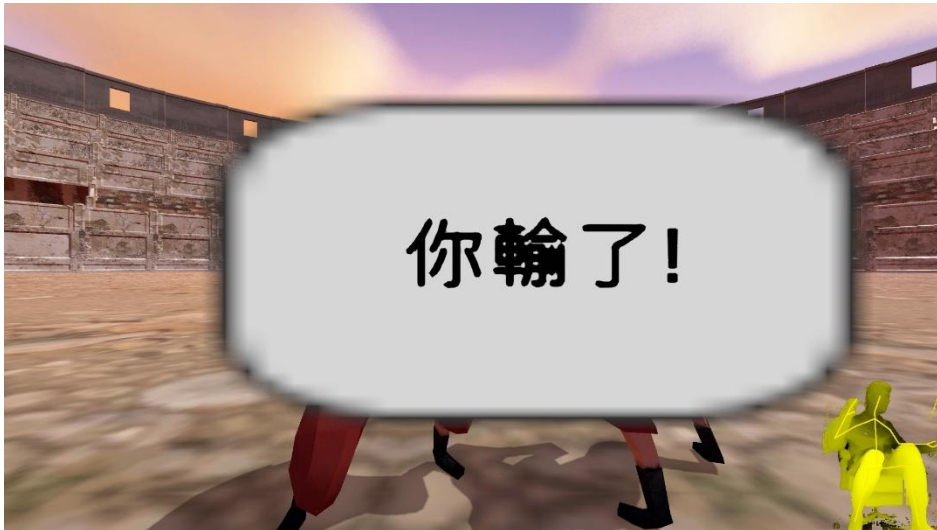


圖 8 mode2 結算畫面



圖 9 mode3 遊戲介面



圖 10 mode3 結算畫面

第五章、專題學習歷程介紹

5.1 專題相關軟體學習介紹

在專題初期，我們對 Lightship ARDK 和 Unity 進行了深入的學習。但經過仔細評估，我們認為 Kinect v2 在動作捕捉的精準度和豐富的開發資源方面更具優勢。考慮到我們的專題需要實現效果，我們最終決定採用 Kinect v2 作為主要感測器，並結合 Unity 的強大遊戲引擎進行開發。這個選擇讓我們能夠更有效地實現專題目標。

5.2 專題製作過程遭遇的問題與解決方法

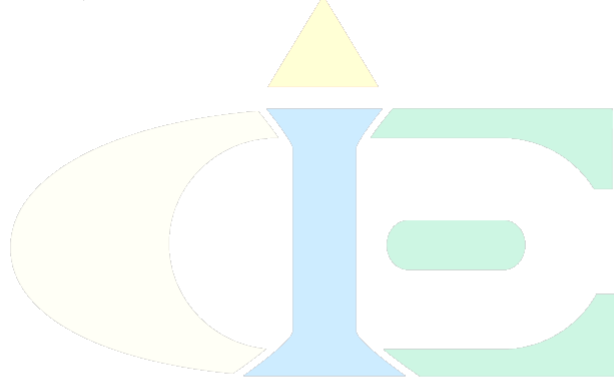
在決定專題方向時，我們一開始打算使用 MediaPipe 來實現體感互動。MediaPipe 的靈活性讓我們可以自定義各種動作辨識模型。然而，在將它的輸出與 Unity 中的骨骼動畫系統整合時，我們遇到了瓶頸。MediaPipe 輸出的關節座標與 Unity 的骨骼結構存在差異，這使得我們在建立映射關係時遇到了困難。經過多次嘗試和調試，但效果都不盡理想。考慮到專題進度和資源限制，我們決定轉向 Kinect v2，因為 Kinect v2 可以直接提供骨骼關節的 3D 座標，並有成熟的 Unity SDK 支援。這個決定讓我們能夠更快速、更穩定地實現我們的目標。

除了技術選型上的調整，專案初期我們在任務分工上也遇到了一些困擾。由於對 AR 遊戲在 Unity 中的實現方式缺乏清晰的認識，團隊成員對於如何分工合作感到迷惘。在深入思考後，我們想到了大學所學的資料結構中的費氏數列。類比於費氏數列的特性，我們將複雜的專案拆解成一系列相對簡單的任務。首先，我們將專案目標分解為「AR 在 Unity 中的實現」、「UI 設計」以及「Kinect v2 的整合」三個主要部分。接著，我們將每個部分進一步細化，例如「AR 在 Unity 中的實現」可以拆解為「場景搭建」、「虛擬物件互動」等子任務。透過這種方式，我們將原本龐大的專案任務化為可管理的小模組，使團隊成員能夠更清晰地理解自己的工作內容，並有效地協同合作。

為了降低開發風險，我們採用了逐步迭代的方式。在完成 Kinect v2 的整合之前，我們先使用簡單的按鈕點擊來模擬體感互動，以驗證整個系統的邏輯是否通順。待 Kinect v2 的開發工作完成後，我們再將按鈕點擊替換為骨骼追蹤，實現真正的體感互動。

第六章、 結論與未來展望

這次的專題讓我們學習在遇到陌生的領域時要如何分工與合作。我們相信，AR 體感運動將為大眾帶來不一樣的拳擊體驗，提升他們的運動樂趣，除了讓心理健康外，或許將來可以因此治療精神疾病。以下是我們對於未來 AR 治療精神疾病的展望，首先是趣味化的運動，AR 運動能讓使用者沉浸在遊戲化的環境中，例如追蹤虛擬物體、完成挑戰任務等，讓運動更具吸引力。可以幫助患有抑鬱症或焦慮症的患者，因為他們常常缺乏運動動力。其次，利用 AR 創造虛擬的人物或群體，模擬社交場合，可以幫助自閉症患者或社交焦慮症患者練習交流技巧。最後是可以減少患者對藥物的依賴，因為運動本身能促進多巴胺、血清素等「快樂激素」的分泌。AR 運動提供了一種非侵入性的方法來改善患者的心理狀態，從而減少對抗抑鬱藥物或抗焦慮藥物的依賴。



資訊工程學系

Computer Science & Information Engineering

參考資料

網頁資料

VR with Andrew: Unity + Kinect Tutorial - Joints (Pt. 2)
(available online at <https://www.youtube.com/watch?v=B7T0XTNk-Vg&t=24s>).

z609054563335: [第二天]從 0 開始的 UnityAR 手機遊戲開發-如何安裝 Unity Hub 與申請 Unity 和 Vuforia 的帳號

(available online at <https://ithelp.ithome.com.tw/articles/10264264>)

Unity Learn

(available online at <https://learn.unity.com/>)

Unity 遊戲開發研究室

(available online at <https://www.cg.com.tw/Unity/>)

yuki: Unity2020 中使用 Vuforia 開發擴增實境 AR APP 教學

(available online at <https://otaru.tw/tutorial/using-vuforia-with-unity3d-making-ar-app-tutorial>)

RoseXiong: Unity 游戏开始界面制作教学

(available online at <https://blog.csdn.net/RoseXiong/article/details/123109434>)

augmented_reality - AR/VR design tools | Prototypr Toolbox | Page 1

(available online at https://prototypr.io/toolbox/augmented-reality-tools/augmented_reality/page/1)