

中華大學資訊工程學系
100 學年度專題製作期末報告



KINECT
for  XBOX 360.

專題：Kinect 體感擴增實境

專題組員：

劉冠麟 B09702084

高震宇 B09702017

指導教授：黃雅軒 教授

專題編號 PRJ2011-CSIE-10003

執行期間 民國 100 年 2 月至 101 年 1 月

目錄

壹、專題計畫摘要	2
貳、開發環境	3
2-1 需求	3
2-2 Kinect for Windows 架構圖	3
參、專題簡介	4
3-1 說明	4
3-2 目標	4
3-3 研究目的	4
3-4 時程規畫	4
3-5 甘梯圖	5
3-6 分工	5
肆、研究步驟	5
※骨架追蹤影像	5
※彩色影像	5
※深度影像	6
伍、成果	10
陸、評估與展望	11
柒、銘謝	12
捌、心得	12
玖、參考文獻	13

壹、專題計畫摘要：

本專題為製作一個能夠辨識肢體模擬鍵盤的應用程式，直覺的對電腦做操作。透過 Kinect 讓電腦辨識肢體形成的軌跡。程式語言選擇的是以 C# 為主，並配合微軟 SDK 的函式庫加以利用。Kinect for Windows SDK 的出現為個人電腦平台帶來了全新的應用程式操作方式，透過 Kinect，不只是遊戲，只要有創意，就可以在 Windows 7 平台上開發出有趣又有商機的 Kinect 應用。體感系統可以將人們透過螢幕方式展現給使用者了解相關資訊。若在個人電腦前使用體感系統，隔著螢幕就可以即時和電腦互動，還可以因為眾多使用方式讚賞，體感系統的未來發展，具有一定的貢獻度。經由本專題設計虛擬互動方式，為學校增添不少色彩。

貳、開發環境：

2-1 需求

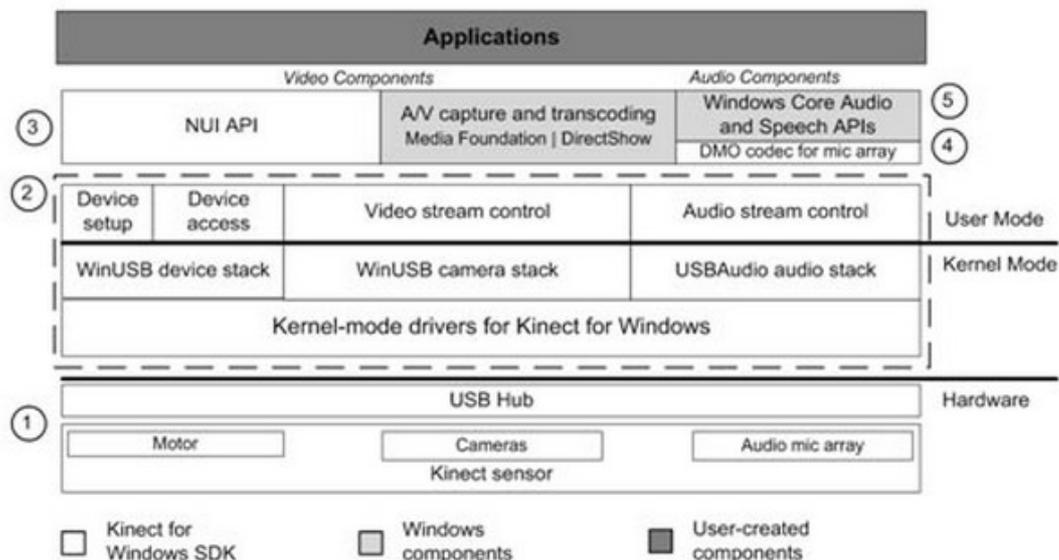
※硬體：

1. 作業系統：Windows 7 (x86 / x64)
2. CPU：雙核 2.66GHz 以上
3. RAM：2GB 以上
4. 顯示卡：支援 DirectX 9.0c 以上
5. Kinect 感應器

※軟體：

1. Visual Studio 2010 或 Visual C# 2010 Express
2. NET Framework 4.0
3. Kinect SDK for Windows

2-2 Kineck for Windows 架構圖



圖一、Kineck 架構圖

Kinect 主要架構分為五大區塊如圖一，以下是詳細介紹：

. Kinect 硬體

這部份指的就是 Kinect 底座的馬達、三個攝影機 (RGB 攝影機、紅外線 CMOS 攝影機、紅外線發射器)、以及陣列式麥克風。

. Kinect 驅動程式

核心模式下包含了以下驅動程式：

- Microsoft Kinect Audio Array Control
- Microsoft Kinect Camera

- Microsoft Kinect Device
- Kinect USB Audio

. NUI API

. 麥克風陣列 DMO 編碼器

. Windows 7 內建影音處理

也就是架構圖中灰色底的部分，這就是為什麼 Kinect for Windows SDK 只支援 Windows 7 的主要原因了。

參、專題簡介

3-1 說明

Kinect 的設備來擷取影像資訊，抓取身體關節各部位以及偵測深遠位置並取得正確的移動資訊，使得操作更順暢。製作一個能夠辨識肢體模擬鍵盤的應用程式，直覺的對電腦做操作，透過 Kinect 讓電腦辨識肢體形成的軌跡。程式語言選擇的是以 C# 為主，並配合微軟 SDK (Software Development Kit) 的函式庫加以利用。

3-2 目標

本專題結合 Kinect 與遊戲軟體發展出一套可用肢體來控制的程式，它結合兩個條件：(1) 即時互動，和 (2) 直覺操控。

3-3 研究目的

在這科技發達的時代，電腦也開始普及化，相信每個人對電腦都不陌生，有人使用電腦上網、玩線上遊戲或使用 SKYPE、MSN、即時通聊天開視訊等等，如此也顯得電腦真的在我們生活中越來越重要了。不過以目前繁多的電腦遊戲裡，利用視訊來進行遊戲執行的 GAME 卻是非常的少，所以我們以視訊為主題，並配合當紅的科技產品 KINECT，創作出與電腦做互動的即時性遊戲，希望如此新穎的視訊遊戲會讓大眾對這種特別的遊戲操作模式感到有興趣。透過這次專題，培養了我們團隊合作的能力，也培養了我們自己實作之能力，未來在投入就業市場時也不怕和其他隊員合作不來，專題的完成，讓我們獲益良多。此專题目的為視訊遊戲，主要的想法是想讓 KINECT 攝影機的使用者可以和電腦做互動式的遊戲，例如使用者可以因藉由視訊攝影到的行為動作來和電腦中的人物下象棋或是打蚊子等等，不過到時候會再因遊戲設計的狀況而增加或修改內容，最終達到我們想要的視訊互動式遊戲。目前我們打算先著手研究遊戲程式的撰寫，所以每個人必須先弄懂學會 VC#，再配合 SDK 既有的函示庫再加上自己所撰寫和鍵盤結合的程式結構，對於剛要入門寫小遊戲的我們幫助很大，之後可能還會使用 OpenCV 來做影像的判別和一些繪圖美工軟體等等。

3-4 時程規劃

本專題進行方式：

初期：一開始先進行專題的構思內容開發，然後確定方向之後，開始尋找並收集

相關的資料與書籍，接下來是規劃專題的進度與步驟，分配工作與討論如何開始並進行製作。

中期: 因為 SDK 為新軟體, 因此在開發上有許多困難, 而市面上的資料也僅限於此, 也因此開發特別困難。

後期: 此時程式架構大致已製作完成, 接著要進行遊戲的整合, 然後進行除錯與測試, 進而讓程式與遊戲之間互動更順暢。

3-5 甘梯圖

	%	%	%	%	%	%	%	%	%	%	%
專題規劃	█										
收集資料		█	█								
準備環境			█	█	█	█					
熟悉軟體操作					█	█	█	█			
設計規劃程式架構								█	█	█	█
除錯測試											█
預計累積進度	10	20	30	40	50	60	70	80	90	100	

表一、甘梯圖

3-6 分工

- ◎ 劉冠麟(組長兼 Designer)
 - › 主要使用者介面、程式操作流程設計
- ◎ 高震宇(Developer)
 - › 程式架構設計及實作開發操作員

肆、研究步驟

本專題的研究步驟分為 A~E 等五項, 即 A: 影像資料, B: 處理影像, C: 鍵盤參數以及使用類別, D: 數學運算, E: 測試程式。

A. 近期來有關 Kinect 心得分享文章, 了解使用 Kinect SDK 相關定義函式, 以下是 Kinect 擷取同一物體影像資料:

骨架追蹤影像: NUI 骨架 API 提供最多兩個玩家的骨架位置與方向的資訊, 應用程式取得的是一堆座標的集合, 稱為骨架關節位置 (Skeleton Joint Positions), 因此在初始化 NUI 時一定要指定要使用骨架資料, 並且啟動骨架追蹤, 如圖二。

CMOS 彩色影像: 影像格式為 YUV, 它是一種 16 位元, UYVY 彩色點陣圖等同於 RGB 影像, 好處是使用較少的記憶體, 也因此只支援到 640x480, 15FPS 的影像, 如圖三。

深度影像：在深度影像的畫格中，每一個像素（16 位元）的資料，表示在指定的 x, y 座標下，與最靠近 Kinect 感應器的物件距離（單位：mm），深度距離範圍 850mm ~ 4000mm，深度 0 則表示未知狀況，如圖四。



圖二、骨架追蹤影像

圖三、CMOS 彩色影像

圖四、深度影像

B. 透過微軟開發 Kinect SDK 追蹤骨架處理骨架座標。



圖五、骨架關節位置 (Skeleton Joint Positions)

Kinect 可以透過 SkeletonData 的 Joints 集合來取得某一個玩家的所有關節的 3 維座標如圖五 20 個關節點，你可以使用 JointID 這個列舉來指定要取得那一個關節的座標

C. 為了連接到鍵盤，需查詢鍵盤 Keyboard 和 NativeMethods 類別並且收集鍵盤對應點參數以及對訊號的處理如：Thread.Sleep(1)。

NativeMethods: 用來放常用應用程式界面 (Windows API Constants), 將 Window

s API 整理好寫在一個 DLL 檔裡面，為了避免凌亂與日後可以使用，把所有會用到的常數都寫在下面這個類別裡面, 如表二為部分指令常數。

KeyeventfExtendedkey = 0x0001;
KeyeventfKeyup = 0x0002;
KeyeventfScancode = 0x0008;
MouseeventfVirtualdesk = 0x4000;
SMXvirtualscreen = 76;
SMYvirtualscreen = 77;
SMCxvirtualscreen = 78;
SMCyvirtualscreen = 79;

表二、為部分指令常數

Thread.Sleep(1): 在指定的時間內暫止目前的執行緒。參數的單位為毫秒數。Keyboard 部分參數表格, 如表三

Move = 0x0001	LeftDown = 0x0002
LeftUp = 0x0004	RightDown = 0x0008
RightUp = 0x0010	MiddleDown = 0x0020
MiddleUp = 0x0040	XDown = 0x0080
XUp = 0x0100	Wheel = 0x0800
Absolute = 0x8000	

表三、Keyboard 部分參數

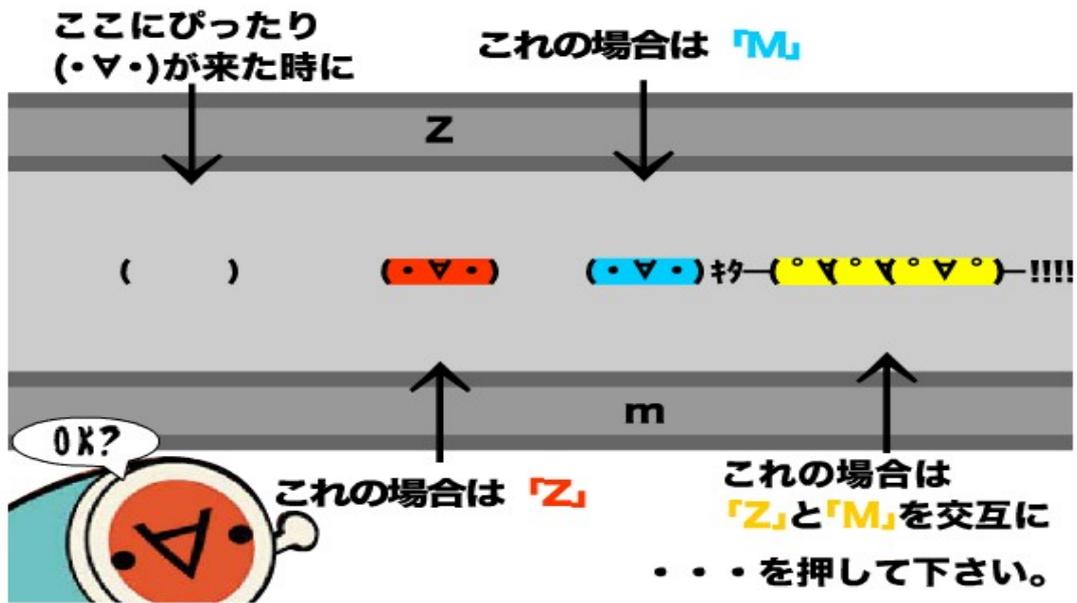
D. 取得的位置後判斷體態與手位置，並且將座標(左手掌與右手掌相減之距離去做判斷), 若是負值則執行某按鍵, 正值執行某按鍵, 同時配合

Keyboard.Type();

控制鍵盤，並且下命令，至於數值方面經過嚴密精控計算後得 6~11 之間和 11 以上的分區做處理效果最佳

E. 測試程式穩定度與實用性，是否符合人體工學知性操作性，並且達到無延遲或不影響操作流暢度。

遊戲畫面放大 (+) 遊戲畫面縮小 (-)



圖六、為遊戲主要介面

控制方法如下：

 相當於按下鍵盤上的 Z

 相當於按下鍵盤上的 M

 相當於輪流按下鍵盤上 Z 和 M



圖七



圖八

圖七、八、為單右手由內往外伸出當作  按鈕來獲得分數

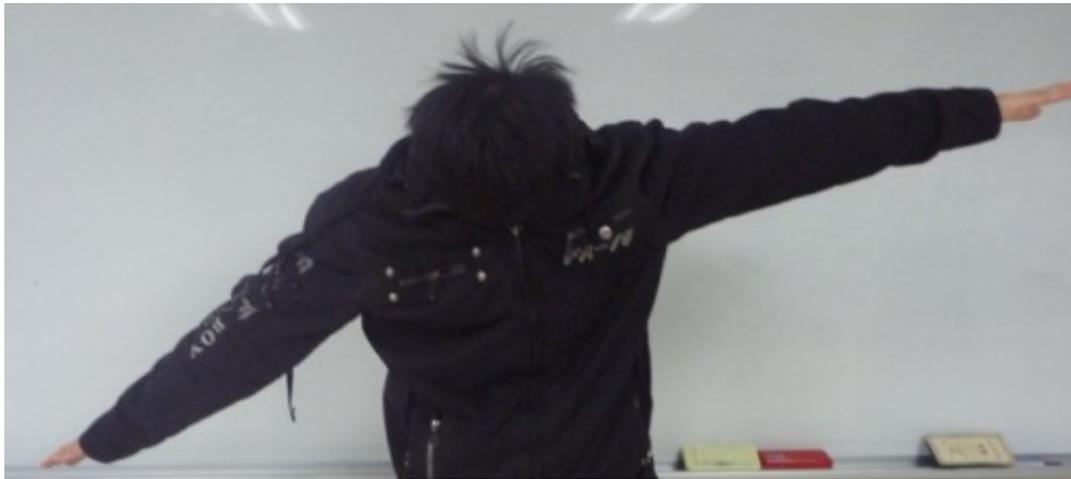


圖九



圖十

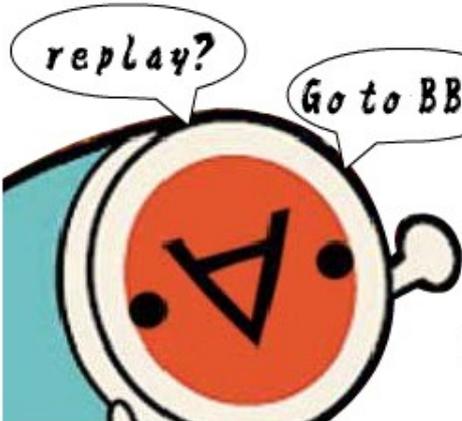
圖九、十、為從單左手由內往外伸出當作  按鈕來獲得分數



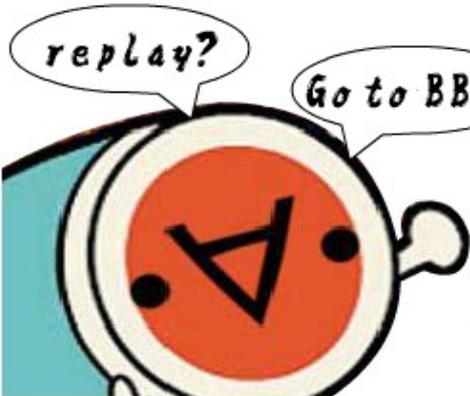
圖十一

圖十一、為雙手臂上下揮動當作  按鈕來獲得分數

伍、成果

	スコア	1350
	最大エロボ	11
	(・▽・)	27
	(・ー・)	45
	(・ω・)	29

圖十二、為初次測試分數
 遊戲畫面放大 ⊕ 遊戲畫面縮小 ⊖

	スコア	30700
	最大エロボ	11
	(・▽・)	45
	(・ー・)	98
	(・ω・)	50

圖十三、為經過多次測試最佳分數

陸、評估與展望

直覺式操作，已是時下科技的趨勢。如微軟 XBOX 新產品 Kinect，使用關節式的操作來控制。而直覺式的好處也在於，人類也不需要再學習多一種操作方式，就可以憑自身的體感操作，已可以達成良好的效果而有鑑於目前的體感操作類產品，數量甚少且其無法應用在個人電腦上，所以我們想藉由 Kinect 的設備來擷取影像資訊，抓取身體關節各部位以及偵測深遠位置並取得正確的移動資訊，使得操作更順暢。在經過組員們長久時間的努力之下，開發這系統的過程中我們發現 SDK 是一套非常方便的軟體，對肢體偵測有興趣的人可以多多使用，因為這套軟體有很多功能都可以直接呼叫函式庫來做運用，例如：專題使用到的技術：偵測深度、骨架追蹤等，這些技術都是直接呼叫函式庫來做撰寫系統，相當的方便。整個系統測試過後，有發現一些錯誤，最嚴重的應該是受到環境影響的問題，如果今天測試的背景不是單一化，光線的明亮度不均勻下，辨識之後的結果會不是預想的結果。之後如果想開發這一類的肢體控制系統，可以運用更深的影像處理讓他更穩定。在這套感應遊戲系統完成時，雖然跟一開始的想法有些出入，不過大部分想要的模式都有做到，然而影像的處理上，仍需要再多加琢磨，或許這樣會使玩家更有興趣對於這套遊戲系統上，希望往後有接手的學弟們可以在影像的效果多作修改。當在這次的專題製作中，我們可以發現到 KINECT 的多廣用途，而不再只是用來玩遊戲的陪襯品，而且視訊幾乎是人人必備的東西，因此利用這方面平台進行的遊戲，更加是另一個遊戲市場的發展，畢竟現在市面上的視訊遊戲幾乎可以說是屈指可數的。然而在許多技術上，當然要好還要更好，因此希望未來可以在平面跟前後取像可以更加順暢，在演算法的內部撰寫能加入相關的程式碼，而在遊戲上可以加入更多的功能，讓玩法不在單一更加有多樣化，我們很看好未來接手此專題的學弟們的發展，希望能讓這份專題變得更有可看性。

柒、銘謝

專題的部份已經過了稍許久的時間了，這段時間非常感謝老師細心的帶領著我們，也時常分享了許許多多寶貴的經驗以及花了不少的時間在督促我們的進度，同時讓我們看了很多的資料，希望能夠透過這次專題的經驗，讓我們在就讀研究所或是就業工作上能有一定的基礎。感謝專題指導教授黃雅軒老師，耐心傾聽我們的意見以及時時講解新的觀念給我們做思考，讓我們設計時遇到的困難不會延宕太久。感謝專題的成員，在這一年裡無私的奉獻時間一同完成本專題。感謝資訊工程學系的全體老師，在這四年學到的知識都是本專題完成的重要關鍵。

捌、專題組員感謝與心得

劉冠麟：

首先，我要先感謝各位同組組員這一年來的忙碌和辛勞，在製作這份專題的途中，我們一路上碰到了許多的問題和障礙，而且能找的相關文獻資料並不多，於是少不了大家互相焦慮所衍生的心情不好，不過有別於其他組的同學，我們組員之間多了相互包容和尊重，也因此我們在製作時少了許多紛爭和抱怨，這是非常讓我感到欣慰的。最後也要感謝雅軒老師，當我們在面臨問題而無法解決時，他就會很巧妙的在旁邊替我們點上一筆，使得我們頓時豁然開朗，有了明確的方向，當我們碰到想法不靈光而毫無創意時，他就會加上許多新穎的創意讓我們有了目標，感謝他這一年來的辛苦。

高震宇：

做完這一次的專題以後，覺得如果真的要做出一個新的程式，非常有挑戰性，而且這項新研究沒有學長姐們研究過，所以去了解 KINECT 的訊號抓取花了一些時間。我們在程式方面也下了不少功夫，因為要讓肢體可以穩定的控制遊戲花了很多的時間。而在視訊的追蹤中我們也發現了不少的難題和困難，比如說會因為背景的關係而導致不穩定或追蹤錯誤等等，而現在再程式的運作方面雖然還是有些不流暢，可能是因為寫的程式太大的關係而導致，但是之後我們還是會繼續去學習來把它用的更好。

玖、參考文獻

1. Kinect for Windows SDK 官方中文開發教學影片

<https://kheresy.wordpress.com/2011/08/25/kinect-for-windows-sdk-chinese-tutorial/>

這一系列影片總共有九段，分別是：

Kinect SDK 介紹與開發環境安裝

Kinect 初始化與彩色影像擷取

Kinect 深度影像擷取與深度計算

透過深度資料辨認玩家位置

玩家任意背景替換應用

骨架追蹤與螢幕座標轉換

繪製人體骨架與多人骨架追蹤

骨架追蹤範例 - 操控光劍與地球

Kinect 聲音錄製

2. 曹祖聖老師的部落格

<http://teacher.syset.com/viewtopic.msp?p=19003#19003>