

東海大學資訊工程學系研究所
碩士論文

指導教授：林祝興 博士

Dr. Chu-Hsing Lin

使用 Kinect 的三維身分辨識及其應用
3-Dimensional Personal Identification and Its
Applications Based on Kinect

研究生：林思縈

中華民國一〇四年七月

中文摘要

隨著人機介面技術的快速發展，過去僅由裝設在身上的感測裝置來捕捉人體動作，至今已逐步轉為透過攝影機來擷取人體動作。在 2010 年發表 Kinect 感測裝置，跟傳統攝影機或者 CCD 不同的地方在於 Kinect 所接收的資料除了二維影像陣列外，也提供了三維立體環境的資訊，不僅有色彩資訊也多了深度的資訊，而且逐步廣泛應用在數位教學、醫療應用、三維動畫等領域中。

早期動作捕捉裝置較為昂貴，因此多半使用一般的視訊攝影來捕捉人體的動作，由於視訊畫面缺乏深度訊息，僅能對二維姿勢作判別，若應用在公開場所，在安全性的考量上則不太適用。

本研究主要以 Kinect 感測裝置為基礎，藉由 Kinect 官方所提供的 SDK(Software Development Kit)來延續早期的動作辨識，利用三維資訊來進行安全辨識，並結合手指控制電腦實現各種應用。藉著人體的三維資訊無法輕易地被複製的原理，以達到系統安全性的保障，具有各種未來應用的潛力。

關鍵詞： Kinect、人臉辨識、骨架辨識、互動、人機介面

ABSTRACT

With the fast development of the man-machine interface technology, human movement captured by sensing devices attached to the human body in the past has gradually captured through cameras nowadays. Different from traditional cameras or charge-coupled device (CCD) cameras, the Kinect motion sensing device, first released in 2010, receives information of the three-dimensional stereoscopic environment addition to the two-dimensional image array. It contains not only traditional color information but also depth information, and has been gradually widely applied in digital education, medical applications, 3D animation and other areas. Motion capture devices were expensive in the past, most of them adopted general video cameras to capture human motion. Due to the lack of depth information on the video screen, human movements were often determined only by two-dimensional body posture information. Also, due to security concern motion detection machines are not suitable for public places. This study uses the Kinect motion sensing device with the official Kinect Software Development Kit for human movement identification. Combined with three-dimensional information for security identification, motions of human fingers are further detected to control computers and implement a variety of functions. Since three-dimensional information of any individual human body cannot be easily replicated, the proposed method attains system security protection and is promised for a variety of applications in the future.

Keywords: Kinect, Personal Identification, Face Recognition, Skeleton Recognition, Interaction, User Interface

致謝

歷經兩年的碩士生涯即將結束，求學階段也將在此告一段落，有太多的人需要感謝，有你們的幫助、陪伴及體諒才得以讓我順利完成學業。

首先，需要感謝的是在研究所時期幫助我良多的指導老師 林祝興老師，提供一個適合的環境與大家共同學習，也願意讓我嘗試結合其他領域來做研究，在老師身上不僅學到相關知識，老師經常與我們分享時事、人生道理及為人處事該有的態度等等，都讓我收穫良多。此外，在老師的協助之下參加了國際性的研討會，與各國的學者們交流，並藉此得到國際化的體驗，都大大的提升了自己的能力。

再來要感謝口試當天前來參加的評審委員，張隆池教授、蔡坤霖教授與賴威伸教授，給予論文諸多的建議與改善，使得此篇論文能夠更加完整與充足。特別感謝劉榮春老師，不厭其煩的幫忙修改所投搞論文中的用詞及格式，使得論文得以更流暢。

另外也要感謝這些日子一起陪伴我的朋友們，兩年的研究所生活中，相處時間比家人還多的實驗室學長、同學及學弟們，給了我課程上的協助及研究上的建議。需要散心時，只要一通電話就出現的老夥伴們，和你們在一起總是可以盡情的大笑大鬧。因為有大家的幫忙與關心，我的碩士生涯更增添了一份精彩。

最後，謝謝我親愛的家人們，雖然一開始的路走的不是很順遂，因為你們給了我無上限的支持與鼓勵，得以順利的完成學業。除了謝謝還想說一句對不起，因為我的任性不成熟，總是讓你們擔心，擁有你們給予不間斷的包容、開導及體諒的我，是最幸福的研究生了。

以此誌獻給我敬愛的家人與老師，及所有幫助過我的人，謝謝。

林思縈 於東海大學研究所 資訊安全實驗室 2015 年 7 月

目錄

| | |
|---------------------------|----|
| 中文摘要..... | 1 |
| ABSTRACT..... | 2 |
| 致謝..... | 3 |
| 目錄..... | 4 |
| 表目錄..... | 6 |
| 第一章 簡介..... | 7 |
| 第二章 背景知識與技術..... | 9 |
| 2.1 Kinect 感測器..... | 9 |
| 2.1.1 3D 深度感測器..... | 10 |
| 2.1.2 RGB 彩色攝影機..... | 12 |
| 2.2 RealSense 3D 攝影機..... | 13 |
| 2.3 感測器的應用..... | 14 |
| 2.4 OpenNI..... | 14 |
| 2.5 人臉識別..... | 16 |
| 2.6 人體姿態辨識..... | 16 |
| 2.7 Kinect 的互動..... | 17 |
| 2.8 手勢辨識..... | 17 |
| 第三章 實作系統架構..... | 18 |
| 3.1 硬體架構..... | 18 |
| 3.2 軟體開發環境..... | 18 |
| 3.3 使用者介面..... | 19 |
| 3.4 系統架構與流程..... | 20 |
| 3.4.1 系統架構..... | 20 |
| 3.4.2 系統流程..... | 21 |
| 3.5 實作方法論述..... | 23 |
| 第四章 研究結果..... | 25 |
| 第五章 結論與未來展望..... | 34 |
| 參考文獻..... | 35 |

圖目錄

| | |
|-----------------------------------|----|
| 圖 2-1 Kinect 感測器..... | 9 |
| 圖 2-2 Light Coding 技術理論..... | 11 |
| 圖 2-3 深度影像..... | 12 |
| 圖 2-4 彩色畫面..... | 12 |
| 圖 2-5 Intel Realsense 3D 攝影機..... | 13 |
| 圖 2-6 OpenNI 架構圖..... | 15 |
| 圖 3-1 使用者介面 I..... | 19 |
| 圖 3-2 使用者介面 II..... | 20 |
| 圖 3-3 系統架構圖..... | 21 |
| 圖 3-4 OpenNI 介面..... | 21 |
| 圖 3-5 系統流程圖..... | 23 |
| 圖 3-6 校正姿勢..... | 24 |
| 圖 4-1 辨識人臉圖..... | 25 |
| 圖 4-2 使用者確認..... | 26 |
| 圖 4-3 正確的使用者..... | 27 |
| 圖 4-4 錯誤的使用者..... | 27 |
| 圖 4-5 選擇指令 I..... | 28 |
| 圖 4-6 選擇指令 II..... | 28 |
| 圖 4-7 OK 指令..... | 29 |
| 圖 4-8 勾勾指令..... | 29 |

表目錄

| | |
|-----------------------|----|
| 表 2-1 Kinect 規格..... | 10 |
| 表 4-1 ROC 可能表列..... | 31 |
| 表 4-2 辨識密碼鎖之可能表列..... | 32 |
| 表 4-3 辨識指令之可能表列..... | 32 |

第一章 簡介

傳統中，對於人員識別的方法，通常是持有類似身分證的證件來做為辨識的依據，其使用者介面也大多是以滑鼠來操作，近年來平板電腦的普及使得螢幕觸控功能漸成主流，更演進到了利用生物特徵來當作辨識的依據，如虹膜、指紋、聲紋...等。

現今的影像處理技術多使用彩色圖片來做為辨識，此方法易受到光線的影響而導致辨識率降低。近年由於 Kinect 攝影機的推出，價格比一般專門用來掃描物體深度資訊的器材便宜，若使用 Kinect 攝影機來擷取人臉及身體骨架的深度資訊來做為人員的辨別，不只可以對抗環境光源的影響還可以提高系統的辨識率。

使用人臉辨識具有非接觸性與方便性的優點，對用戶的干擾也比較少，容易被使用者接受，在進行人體骨架的偵測時，可依據骨架變化來判斷人體的動作。此外還可以依照手指的姿勢來進行手勢的偵測，藉以達到更細微的判斷，再按照手部姿勢的不同做出相應的反應。2011 年 6 月微軟公司公布了 Kinect SDK for Windows 的細節，開發者可以使用 C++ 或 C# 來做開發，這無疑是提供了一個平台可供使用。由於開發程式的公開，可以更容易的結合各個領域進而發展出不同的應用。

本論文使用 Kinect 感測器作為影像擷取裝置，試著提出一個安全的辨識系統並結合手勢判別來執行系統，以完成即時辨識系統的應用。此方法有辨識安全性高的優點，且不需配戴辨識工具以減少實用上的不便。系統啟動後，使用者無須配戴任何感測器，只要進入感測器的偵測範圍內，系統將會即時進行人體辨識，也可以進行簡易操作，並將辨識後的結果回傳到人機介面上。由於人體骨架有不可被複製的特性，即使在人臉辨識的部分使用了平面照片而通過辨識，在骨架辨識這一關也會被阻擋下來，使用了兩種辨識因子會使得系統安全性更為提升。而

此系統應用於不適合配戴辨識裝置、穿著特殊衣物、須執行操作且重視安全性的場所，例如工廠、醫院...等。

本論文共分為五章：本章為簡介，說明研究動機與目標；第二章中背景知識與現有技術的文獻探討；第三章在討論系統實作的架構，以及辨識方法的流程；第四章為研究結果；第五章為結論與未來展望。

第二章 背景知識與技術

2.1 Kinect 感測器

Kinect 感測器[1][2]由 PrimeSense 與微軟合作開發(如圖2-1)，機身上有3顆鏡頭，中間的鏡頭是一般常見的RGB彩色攝影機，主要是用來辨識玩家身分，例如人臉辨識和身體特徵，以及辨識基本的臉部表情；左右兩邊鏡頭則分別為紅外線發射器和紅外線CMOS攝影機所構成的3D深度感應器[2][3]，Kinect主要就是靠3D深度感應器偵測玩家的動作。

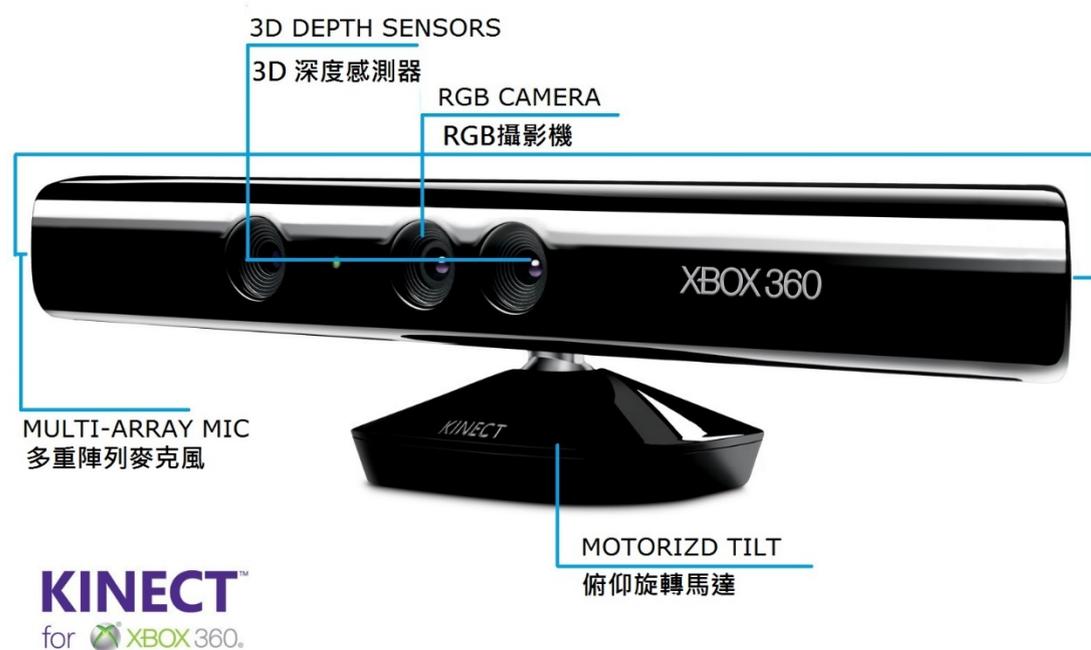


圖2-1 Kinect 感測器 (圖片來源：Kinect for Windows[27])

主要應用於電視遊戲以及娛樂，也能應用在擴增實境遊戲以及視訊通話，玩家在電視機前不需配戴任何物件，即可使用身體的擺動來操作系統與遊戲。其方法為骨骼化技術 (skeleton technology) 可辨識出目前人體的動作。此外Kinect還搭配了追焦技術，若玩家超出影像畫面，底座馬達會隨著玩家的移動跟著跟著水平轉動。Kinect也內建了陣列式麥克風，可藉由多組麥克風同時收音，比對後消

除掉雜音，等於提供了降噪功能，讓玩家的聲音能夠更清楚的傳遞出去。Kinect 規格如下表2-1所示。

表2-1 Kinect規格

| 視野角度 | |
|----------|-------------------------|
| 水平視野 | 57 度 |
| 垂直視野 | 43度 |
| 實體俯仰傾斜範圍 | ±27 度 |
| 深度感測器範圍 | 1.2m~3.5m |
| 資料串流 | |
| 深度感測器 | 320×240 16-bit @ 30 FPS |
| 彩色攝影機 | 640×480 32-bit @ 30 FPS |
| 聲音規格 | 16-bit @ 16 kHz |

2.1.1 3D 深度感測器

3D 深度感測器包含一組紅外光發射與接收器，分別位於 Kinect 的左右兩側，左側為紅外線的發射端，右側則為接收端，利用 USB 埠連接至電腦。

與微軟合作的 PrimeSense 公司是提供動作感測系統的公司，擁有 PrimeSensor 及感測晶片 PS1080，而 Kinect 使用的 3D 偵測技術則是使用 Light Coding 技術[4]。Light Coding 技術理論是利用連續光對測量空間來進行編碼，透過感應器讀取編碼的光線，交由晶片運算進行解碼後，產生成一張具有深度的圖像[5][6]。此項技術的關鍵是雷射光散斑(laser speckle)，當雷射光照射到粗糙物體後，會形成隨機的反射斑點，稱之為散斑。散斑的高度隨機性會隨著距離的不同而變換圖案，空間中任何兩點的散斑皆是不同的圖案，等於是將整個空間加上了標記，所以任何物體進入此空間或是移動時，都可確切的紀錄物體的位置，如

圖 2-2 所示。

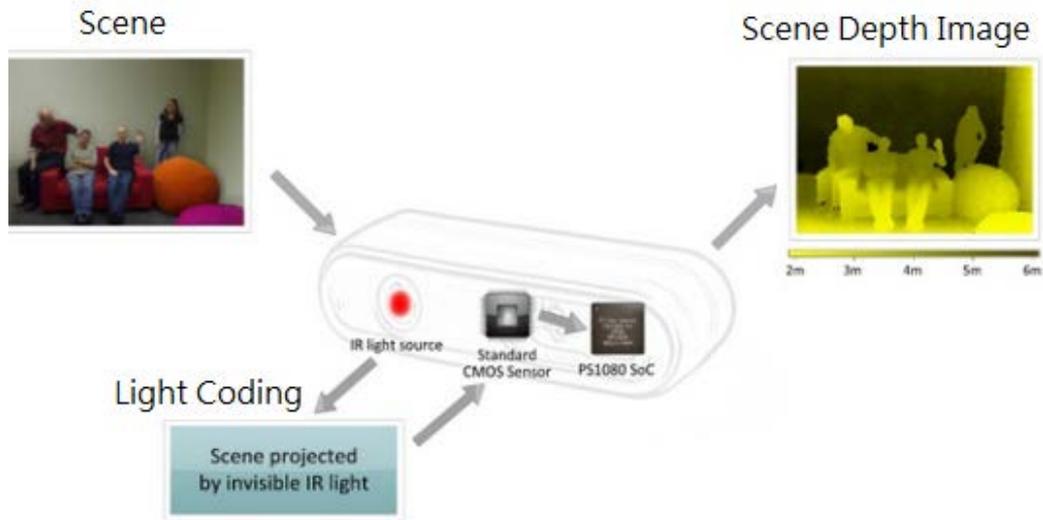


圖2-2 Light Coding技術理論 (圖片來源：PrimeSense[28])

Kinect 就是以紅外線發出人眼看不見的 class 1 雷射光，利用鏡頭前的擴散器(diffuser)將雷射光均勻分佈投射在測量空間中，由紅外線攝影機記錄下空間中的每個散斑，擷取原始資料後，再透過晶片計算成具有 3D 深度的圖像。

微軟會將偵測到的 3D 深度圖像，轉換到骨架追蹤系統。此系統最多可同時偵測六個人，卻只能同時辨識兩個人的動作。其中兩個被辨識的人可記錄 20 組節點，其軀幹、四肢、手指皆為追蹤之範圍，達成全身體感操作。欲了解使用者的動作，微軟也使用了機器學習技術 (machine learning)，來建立龐大的圖像資料庫，以達成智慧辨識能力，盡可能理解使用者的肢體動作所代表的涵義。

深度顯示圖中，利用三原色不同的混合比例，來表示距離Kinect 遠近的不同，如下圖2-3。Kinect 無法量測某些物質與 Kinect 之間的距離，如光滑金屬、吸收紅外光材質之物體、液晶電視顯示面...等，因此在深度顯示圖中以黑色來顯示。



圖2-3 深度影像 (圖片來源：自行拍攝)

2.1.2 RGB 彩色攝影機

RGB彩色攝影機位於 Kinect 感測器中間部分，主要目的為擷取彩色影像至使用者介面。彩色攝影機利用 USB 埠連結電腦，配合 Kinect 函式庫，經由軟體的擷取便可得到彩色影像畫面。彩色畫面將如下圖2-4 所示。



圖2-4 彩色畫面 (圖片來源：自行拍攝)

2.2 RealSense 3D 攝影機

為了讓人與科技之間的互動更簡易、自然且融入，英特爾(Intel)公司與其他廠商合作，透過使用 Intel RealSense 技術的新硬體與軟體產品線，將模擬人類感知的功能融入到各種內含英特爾技術的裝置。擁有效能頂尖的景深感測器以及彩色攝影機，可偵測手指的細膩動作，並具備高精準度的手勢辨識與人臉五官偵測功能，藉以了解人體動作與情緒。它能在前景與背景模式下執行認知功能，以進行操控、強化互動擴增實境，掃描三維物體影像或執行其它功能。此款 3D 攝影機也是全球第一款結合 3D 景深與 2D 攝影機的模組，協助裝置「看到」和人眼所見一樣的景深。將會整合於各式裝置中如圖 2-5，包含二合一、平板電腦、Ultrabook 及 all-in-one 等等[30]。



圖2-5 Intel Realsense 3D攝影機 (圖片來源：Intel [30])

RealSense 3D 攝影機採用深度感測技術，讓您的電腦觀看世界的方式更像人類，主要的特性有以下幾項：

- 根據需求打造自己的聊天園地。
- 揮手即可控制裝置。
- 掃描、列印和分享完全以 3D 呈現。
- 感測深度並追蹤人體動作。

2.3 感測器的應用

對於現實生活上的觀測或記錄，首先就會想到感測器，感測器的種類繁多，能感測許多種類的狀態變化，如聲音、位置、溫度、濕度及加速等等[9]。隨著遊戲產業的逐步演進與興盛，因為其成本低廉及優異的動作辨識能力，感測器也逐漸被廣泛應用於各種低成本之體感互動系統及遊戲當中，任天堂在 2006 年推出的 Wii 便是一個感應器應用在娛樂上的例子[7]。其感應器便是運用在 Wii 遙控器 (Wii Remote) 以及感測條 (Sensor Bar) 上，結合了以上兩種控制器之後，便可以做出許多不同於以往的遊戲方式[8]。

使用感測器所偵測出的這些變化中，若將與人體動態有所關連的部分記錄下來，便會成為另一種生活上的記錄，舉例來說：使用加速感應器即可計算走路的步數，或是運用 GPS 來定位便可了解目前的所在位置，或是利用感應器做為輔助的學習系統...等等的運用[10][11]。

2.4 OpenNI

OpenNI (Open Natural Interaction) 可以翻譯成開放式自然操作，OpenNI 對自然操作 (Natural Interaction, NI) 的定義包含了語音、手勢、身體動作等，基本上就是屬於一種直覺式、且使用者身上不需要其他特殊裝置的操作方式。

OpenNI 定義了撰寫自然操作程式所需要的 API，提供一個多語言且跨平台的框架，藉此提供了一個標準的介面，讓程式開發者在使用視覺、聲音相關感應器，以及分析資料的中介軟體 (middleware) 時，可以更為方便。

圖 2-6 為 OpenNI 之架構圖，基本上分為三層，最上層是應用程式 (application)，也就是程式開發者自己要撰寫的部分；最下方的一層則是硬體的部分，目前 OpenNI 支援 3D Sensor、RGB Camera、IR Camera、Audio Device

這四類硬體。中間這層就是 OpenNI 的部分，主要是負責和硬體的溝通並在自身內部預留了加上中介軟體 (middleware) 的空間，可以用來做辨識及追蹤的處理。

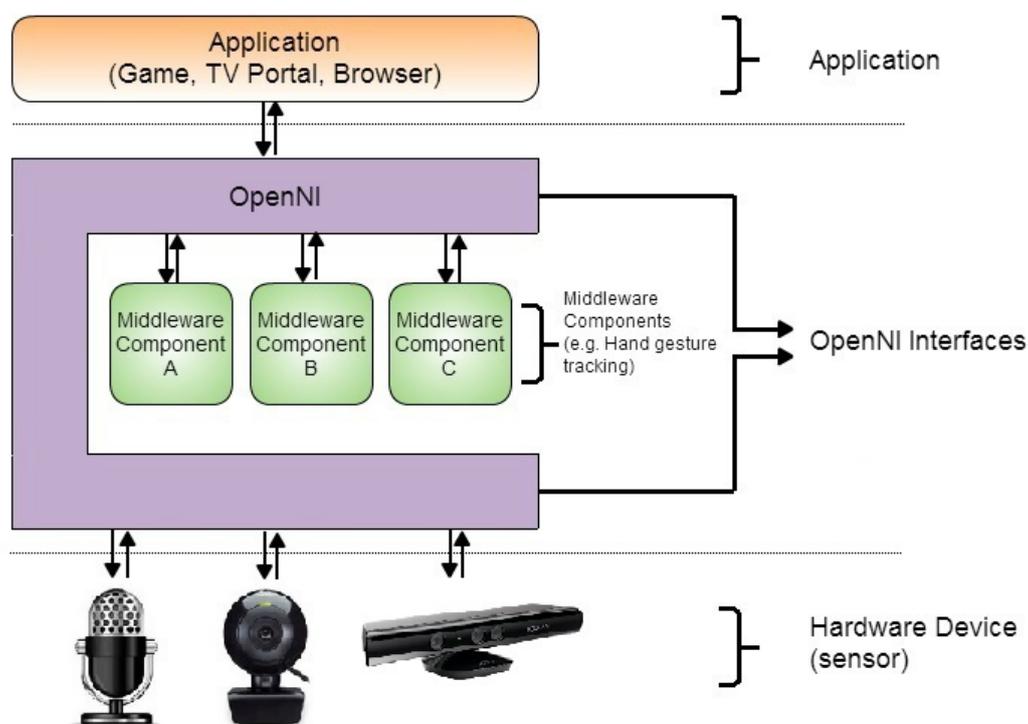


圖2-6 OpenNI架構圖 (圖片來源：Heresy Space [29])

OpenNI 目前在 middleware 的部分，定義了下面四種元件：

- 全身分析 (full body analysis)：由感應器取得的資料，產生身體的相關資訊，例如關節、相對位置與角度、質心位置等。
- 手部分析 (hand point analysis)：追蹤手的位置。
- 手勢偵測 (gesture detection)：辨識預先定義好的手勢，例如揮手。
- 場景分析 (scene Analyzer)：分析場景內的資訊，例如：分離前景和背景、地板的座標軸、辨識場景內的不同物體。

目前 PrimeSense 也已經提供了一套 NITE 當作最主要的 middleware，可提供上述所列的功能。

2.5 人臉識別

人臉識別的相關研究已經發展多年，由於早期受到設備的限制，多半都是二維影像的辨識方法，後續也提出許多改善辨識率與效能的研究。硬體的進步與影像擷取設備的普及化，許多研究也從早期的二維影像進步到使用三維影像進行辨識[12]。三維人臉辨識是指將偵測到的三維臉部數據與資料庫中的三維臉部資料做比對，及可辨識對象身分[13]。為了要減少三維追蹤所產生的誤差，則利用 RGBD 照相機(即 Kinect 感測器)與 AAM fitting 方法來運作。即時的臉部追蹤和識別也由兩種方法所組合而成的框架來完成[14]。

2.6 人體姿態辨識

單純使用影像辨識人體姿態又分為兩種方法，一是使用背景相減的方式，將輪廓與背景區分開來，使用一些基本的條件來辨識人體姿勢，條件包含站立、跪姿、彎腰、蹲下及坐姿...等等[15][16][17]；另外一種則為骨骼抽取法(skeleton extraction)，先將人體形狀進行骨骼化(即細線化)，並設定起點、終點、線段長度與斜率，便可以每條線段來代表人體的不同部位，接著分別標示出每一節點及線段，計算(X, Y)之座標，利用這些資訊來設定不同的姿勢[18]。

另外，三角化切割法(Delaunay triangulation technique)[19][20]也常用於辨識人體姿態，首先取人形影像的四個端點及中央點，將三角形結構細分為更小的三角形，計算每個三角形的重心並連接起來，可以得到人體的細線化資訊以判斷其姿勢。圖上些微變動只會影響區域三角形，對整張影像來說，還是可以辨識出來，並不會有太大影響。

2.7 Kinect 的互動

Kinect 的互動系統中，由 Kinect SDK 或是 OpenNI 取得使用者的全身骨架資訊，並以骨架資訊分析出使用者的肢體動作行為，最後執行相關之指令。為了使 Kinect 可以取得完整的使用者肢體以便分析骨架座標，使用者必須在距離 Kinect 約三公尺處操作系統，導致此互動系統有較大的空間需求。

欲提供社會福利給活動受限之人，線上健身房的概念已被提出，使用者可以利用微軟的 Kinect 來參與網路上的健身活動並進行互動[21]。許多種應用程式若增加了互動此項目，操作更為容易且可以吸引更多人來使用，像是使用手部動作即可操控博物館內 3D 模型建造出來的考古文物[22]。

2.8 手勢辨識

在娛樂與遊戲系統的快速發展下，給我們的生活帶來了更自然且更直接的人機介面。雖然開發人員已明確的定義一組身體各個部位的關節位置，但還是缺乏了強大的機制來執行高級手勢辨識，所以手勢辨別依舊耗時，基於機器學習演算法中，提出了 EasyGR(Easy Gesture Recognition)，為一項助於減少手勢識別程序的工具[23]。

建立一個可結合 Kinect 與高清網路攝影機和投影機的系統，來獲取使用者的手部運動，該系統可廣泛用於公共場所和不允許使用攝影設備的環境[24]。透過被偵測到的指尖區域進而從圖像裁剪後做為目標圖像儲存起來，便可利用儲存的圖像來進行互動。

第三章 實作系統架構

在本章將詳細介紹組成系統所需具備的軟硬體設備與系統流程。硬體架構部分：將說明程式運作所需的電腦、擷取人形影像所需的 Kinect 感測器等硬體設備規格。辨識系統所需的軟體開發環境部分：將說明系統應用的軟體與函式庫、顯示辨識結果的使用者介面，以及顯示人體在不同姿態的特徵參數的開發者介面。

3.1 硬體架構

本硬體架構包含兩個部份，分別為電腦端與 Kinect 感測器端。電腦端接收感測器端的深度資訊後，進行姿勢辨識程序；Kinect感測器端與電腦端利用 USB 埠連接，負責輸出深度資訊與彩色資訊；電腦端接收感測器端的彩色資訊經過電腦端處理，顯示目前房間內彩色影像與人體影像。至於個人電腦環境，本系統是使用桌上型的電腦，CPU規格為Intel® Core™i5 3.2GHz，作業系統使用Windows 7 64位元，記憶體為4GB RAM。

3.2 軟體開發環境

軟體開發的環境主要是以下面三項軟體及函式配合開發：

- (1) Microsoft Visual Studio：此軟體為目前普遍使用設計介面，可以搭配其他使用者開發的函式庫與資料庫配合使用。
- (2) OpenNI 函式庫：為「Open Natural Interaction」的縮寫，大致上可以翻譯為開放式自然操作。OpenNI 對自然操作的定義包含了語音、手勢、身體動作...

等等，基本上就是比較直覺且操作者身上不需要其他特殊裝置的操作方式。

(3) Libfreenect 與 Libusb 函式庫：Libfreenect 是由網路上 Joshua Blake 為主所撰寫出來的 Kinect 應用函式庫，當中包含了 RGB 影像的擷取、深度資訊擷取、LED 燈閃爍控制、Kinect 鏡頭俯仰擺動之馬達控制...等[25]。由於電腦無法直接解析 Kinect 的訊號資料，因此需配合 Libusb 函式庫來解析，解析後電腦即能順利的讀取與傳送訊號至 Kinect[26]。

3.3 使用者介面

主要目的為顯示目前辨識的狀態結果，主要包含了以下部分：

顯示影像：此畫面顯示辨識後的影像，如圖3-1所示。

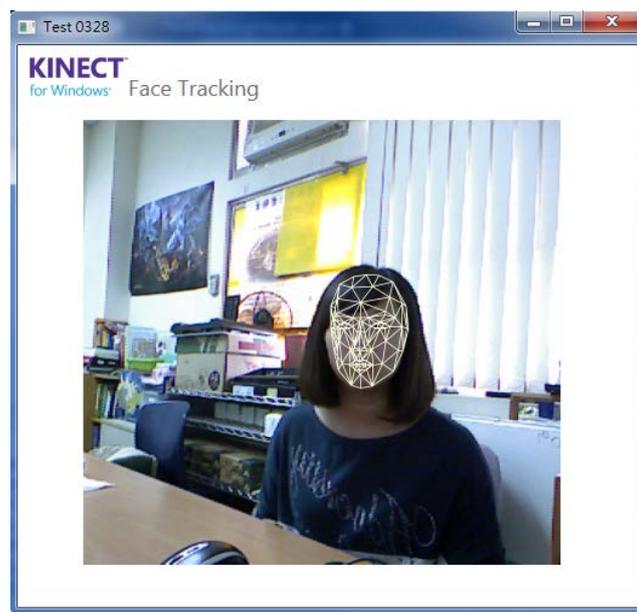


圖3-1 使用者介面I(圖片來源：自行拍攝)

辨識結果：此畫面顯示辨識影像後的結果，結果顯示”Success”或”Failure”，如圖3-2所示，顯示”Success”才可進入下一步驟。



圖3-2 使用者介面II(圖片來源：自行拍攝)

3.4 系統架構與流程

此小節說明本系統之系統架構與流程，第二小節說明本系統最主要的三種辨識功能，分別為臉部偵測、骨架辨識以及手部互動指令。

3.4.1 系統架構

本系統之系統架構如圖3-3所示，主要包含：辨識介面(recognition interface)、指令分析系統(command analysis component)，以及展示系統(display component)三大部分。在辨識介面部分，系統利用 OpenNI 在中介軟體所定義的元件中如圖3-4所示，取得使用者的臉部座標及全身骨架座標資訊。將取得的使用者臉部座標及全身骨架資料進行辨識，並將所需之關節群座標資料取出，計算其座標、方向向量等資料，進而得知使用者是否為此系統操作之允許者。

指令分析系統則分析由辨識介面所取得之各關節群的座標資訊，與觸發指令之動作進行匹配，比對身體動作之辨識結果，接著利用手掌區塊之互動來執行相

對應的指令，最後由展示系統顯示互動結果。

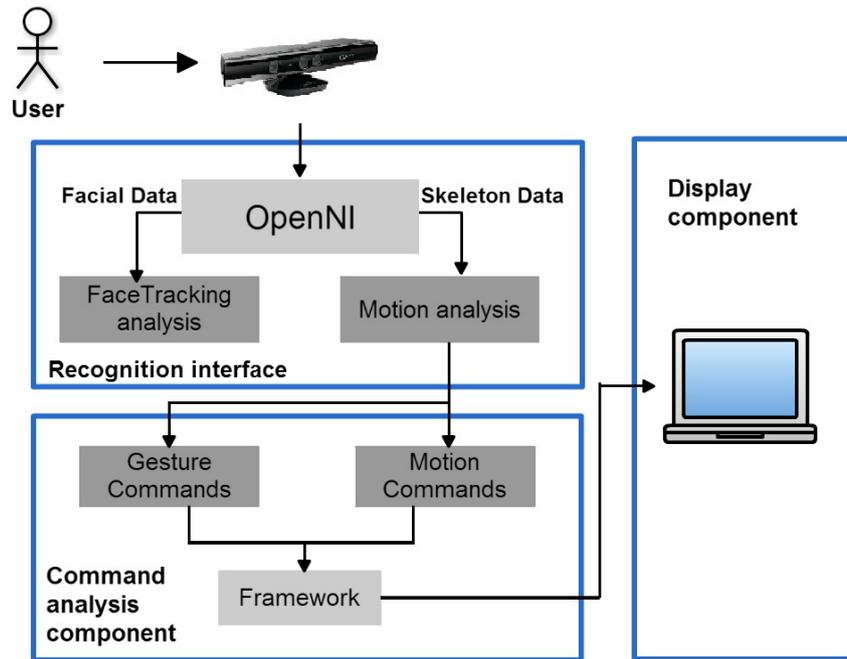


圖3-3 系統架構圖 (圖片來源：自行設計)

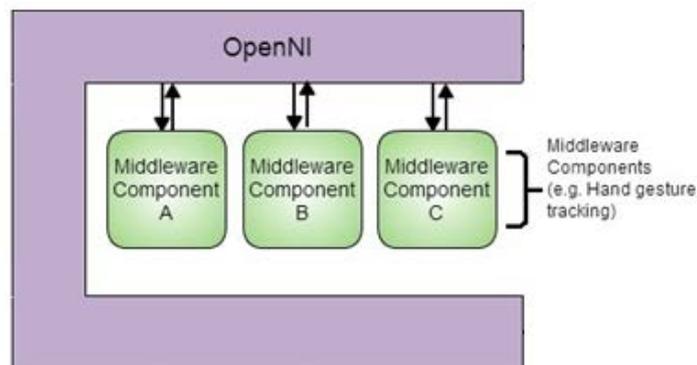


圖3-4 OpenNI 介面 (圖片來源：Heresy Space [29])

3.4.2 系統流程

本論文利用 Kinect 感測器作為影像擷取裝置，只要進入感測器的偵測範圍內，系統將會即時進行人體辨識，通過臉部以及骨架這兩個關卡的辨識，才可進入第三步驟，利用手勢的不同來進行簡易的互動功能，並將辨識後的結果回傳到人機介面上。

系統主要的辨識流程如圖 3-4 系統流程圖，若是首次使用此系統，則必須先在臉部偵測、骨架動作以及手勢指令中建立資料庫讓系統允許此使用者與設定之動作，即為下圖之②、④、⑥；若非首次使用系統，就可以遵照流程之①、③、⑤、⑦來執行系統，其詳細說明如下：

①：機器偵測到使用者後會自動對臉部範圍來進行辨識，經資料庫確認後，若為允許的使用者則進行到下一步驟，若為非允許的使用者則退回系統最初的畫面。

③：系統會分析使用者擺出的骨架動作與資料庫中設定好的密碼鎖動作，若擺出的骨架動作與密碼鎖動作相符便可進行到下一步驟來與系統做互動，若不符則退回系統最初的畫面。

⑤：當手勢指令資料庫與使用者的手勢進行配對後，若配對失敗則退回系統最初的畫面，若手勢配對成功即可執行指令。

⑦：在執行指令中設定了一個手勢當做結束的指令，當執行到結束指令時則此系統將會結束，否則便可繼續執行其他指令。

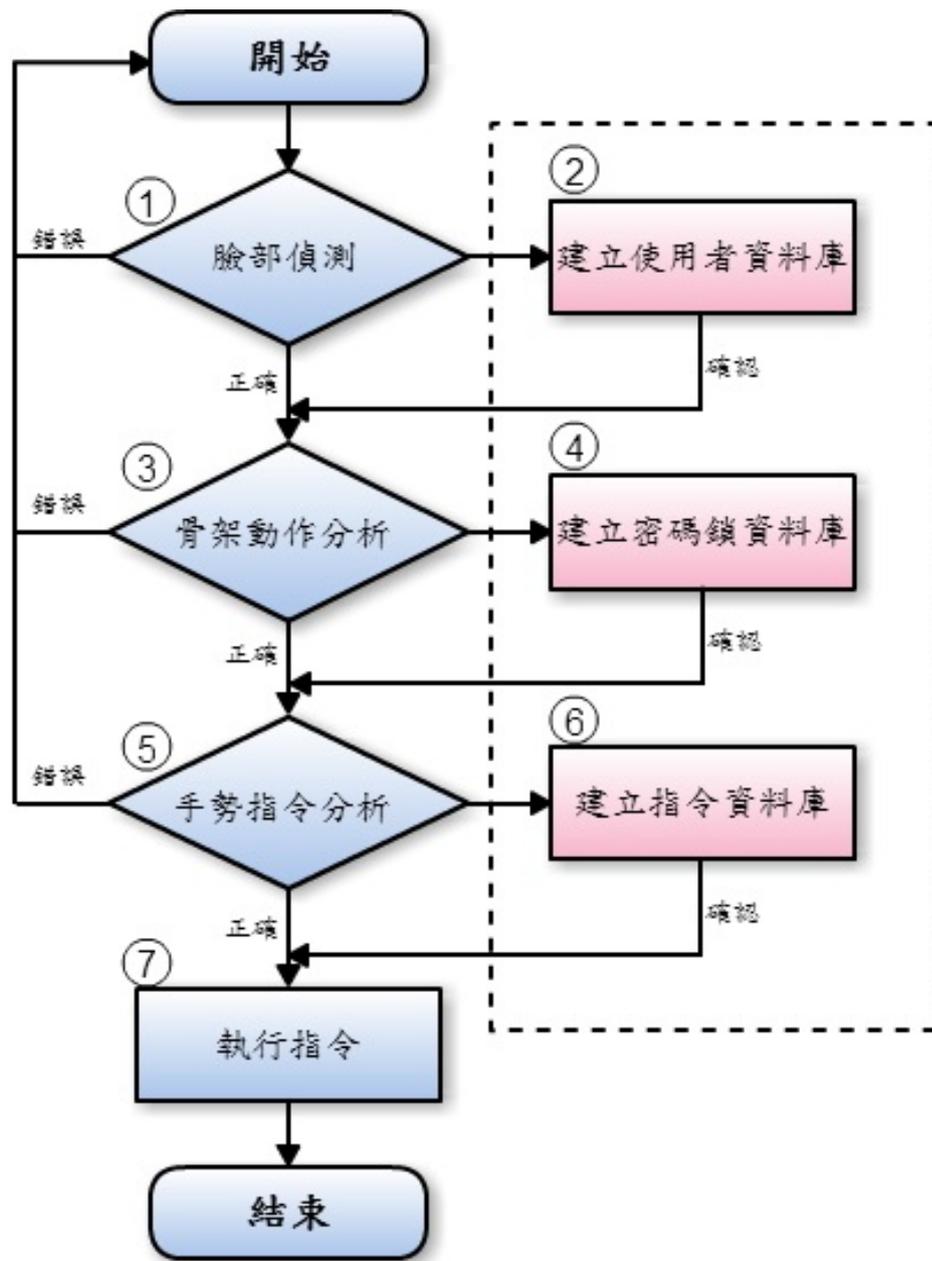


圖3-5 系統流程圖

3.5 實作方法論述

在第一次做設定時，利用 Kinect for Windows SDK 來偵測彩色影像，當偵測到影像中有人臉時，系統將會自動把偵測到的人臉的座標資訊記錄起來，當作是允許的使用者，才可以進行下一階段的識別。

當 Kinect 偵測到新的骨架座標時，會要求使用者擺出校正姿勢，如圖 3-5 所示，這個動作是為了要讓 Kinect 記錄此使用者的骨架座標資訊，校正姿勢只有在出現新的骨架座標時才需重新設定，接著利用身體擺出欲設定為密碼鎖的姿勢，讓 Kinect 記錄此使用者。若沒有設定校正姿勢或出現不同的使用者時，系統將無法成功配對，進而無法通過此步驟的驗證。



圖3-6 校正姿勢 (圖片來源：自行拍攝)

利用骨架關節定位出手部區域範圍，擷取該區的深度影像來做處理，並利用座標來得到該像素的色彩值，`bitmapData.threshold` 函式可以保留手部輪廓範圍，接著將得到的手部輪廓做模糊化，這麼做是為了要去除雜訊，當雜訊去除後取出輪廓線，即可做手指偵測了，利用設定好的手部姿勢來執行系統，考慮到整個系統的複雜度，所以在設定的部分加入簡單的執行指令。

由於 Kinect 也用到了機器學習技術 (machine learning) 當以上的設定都完成後，允許的使用者在往後每一次的使用上都無須再做設定，即可直接使用。

第四章 研究結果

研究結果分為辨識的三個步驟，分別是臉部辨識、骨架辨識及手部互動指令。每一步驟是相連且按照順序的，需成功辨識第一步驟後才可接續第二步驟的辨識，完成兩個步驟的辨識後，便可使用互動指令來簡易操作此系統。

第一步-臉部辨識

利用微軟所提供的 Kinect or Windows SDK 可以進行人臉的追蹤，使用彩色影像、深度影像及骨架資訊來偵測人臉，並將人臉加入了骨架偵測，取得座標資料以增加準確率。當 Kinect 偵測到人臉時，如圖 4-1，系統會利用資料庫檔案自動比對使用者，確認使用者後即可通過此步驟的驗證，如圖 4-2；若偵測到是新的使用者，則會要求使用者做設定。

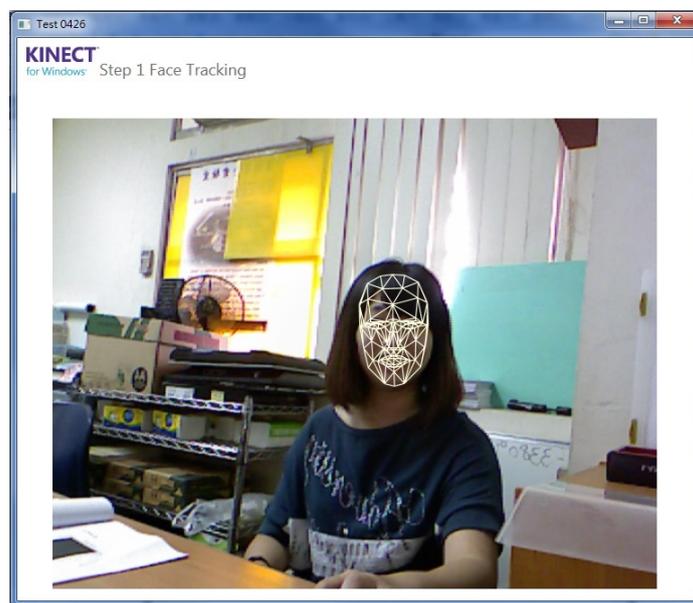


圖4-1 辨識人臉圖

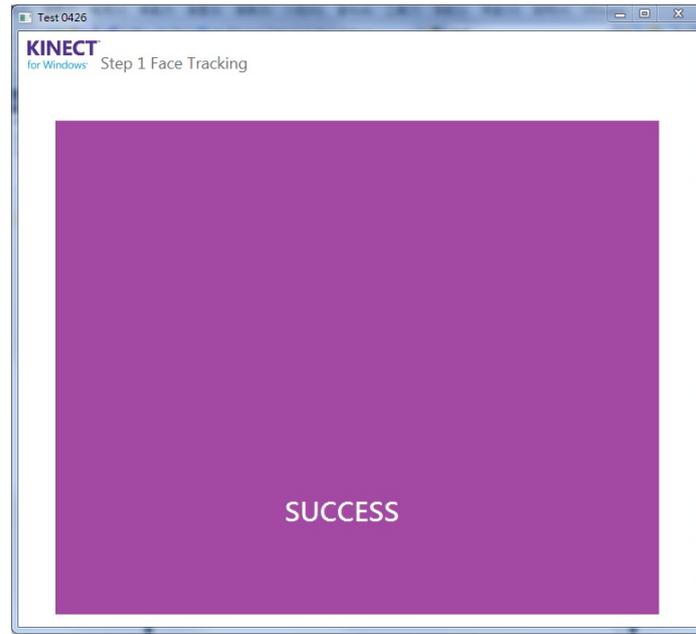


圖4-2 使用者確認

第二步-身體骨架辨識

透過 Light Coding 技術獲得的只是基本的影像資料，主要還是要將辨識到的影像轉換為動作指令。骨架偵測到新使用者時，會要求擺出校正姿勢，進而將此使用者的骨架座標資訊儲存，當使用者再次操作時，系統會讀取特定使用者校正過後的骨架資料，透過 LoadCalibrationData 函式讀取出來，再套用到指定的使用者上直接開始追蹤骨架，擺出設定的密碼鎖姿勢時，系統辨識成功即可進入第三步驟，如圖 4-3 所示。若沒有設定校正姿勢或出現不同的使用者時，如圖 4-4 所示系統將無法成功配對，進而無法通過此步驟的驗證。



圖4-3 正確的使用者



圖4-4 錯誤的使用者

第三步-手部互動指令

當臉部及身體骨架辨識成功後，開始與 Kinect 互動。當本系統偵測到手部姿勢時，將會用紅框表示，與手部相關的各項指令分述如下：

- (1) 選擇指令：系統經由分析手部座標，如圖 4-5，比出手指 1 即為選擇 A 指令，如圖 4-6 比出手指 2 即為選擇 B 指令。



圖4-5 選擇指令I



圖4-6 選擇指令II

- (2) OK 指令：系統經由分析手部座標，如圖 4-7 為 OK 指令即是確認的意思，選擇的指令將會送出，並執行。



圖4-7 OK指令

- (3) 勾勾指令：經由分析手部座標，如圖 4-8 為勾勾指令即是退出的意思，執行此項指令則會退出系統；若不慎執行，則須從第一步重新開始驗證身分。



圖4-8 勾勾指令

由於 Kinect 攝影機的影像更新頻率為 30FPS，代表動作傳遞會有 33ms(1/30 秒)的延遲，因為 PrimeSense 公司所提供的處理晶片只能處理 VGA 畫質、30FPS 的影像。但這個延遲其實對人類造成的影響不大，因為人類對事物的反應速度幾乎都超過 100ms，對於 30FPS 所帶來的 33ms 延遲並不會有影響。

而實際去執行此系統後，針對 Kinect 攝影機的延遲也作了實驗，臉部辨識的影像更新頻率的約為 20 FPS，代表動作傳遞會有 50ms (1/20 秒)的延遲，身體骨架辨識的影像更新頻率的約為 18 FPS，代表動作傳遞會有 55ms (1/18 秒)的延遲，而手部互動指令的影像更新頻率的約為 15 FPS，代表動作傳遞會有 66ms (1/15 秒)的延遲，但對於人類的反應速度來講，這些也是不會有明顯延遲的情況產生。

為了表示此系統的偵測準確率，使用了 ROC (Receiver Operating Characteristic)做為分析工具，ROC 是一種二元分類模型，也就是輸出結果只有兩種類型的模型，例如：陽性/陰性、接受/拒絕、有病/沒病、垃圾郵件/非垃圾郵件...等等。數十年來，ROC 分析被用於醫學、無線電、生物學、犯罪心理學領域中，而且最近在機器學習(machine learning)和數據挖掘(data mining)等等的領域也得到了很好的發展。

二元分類模型的個案預測有四種結果[註]：

真陽性(True Positive)：預測為正確使用者，實際也為正確使用者。

偽陽性(False Positive)：預測為正確使用者，實際卻為錯誤使用者。

真陰性(True Negative)：預測為錯誤使用者，實際也為錯誤使用者。

偽陰性(False Negative)：預測為錯誤使用者，實際卻為正確使用者。

[註]真陽性(True Positive)的比率又稱正確接受率，偽陽性(False Positive) 的比率又稱錯誤接受率，真陰性(True Negative) 的比率又稱正確拒絕率，偽陰性(False Negative) 的比率又稱錯誤拒絕率。

這四種結果可以畫成 2×2 的可能表列(contingency table)，如下表 4-1，假設 P 為陽性(Positive)，N 為陰性(Negative)：

表4-1 ROC可能表列

| | | 真實值 | |
|-----|----|-------------|-------------|
| | | P | N |
| 預測值 | P' | 真陽性 (TP) | 偽陽性 (FP) |
| | N' | 偽陰性 (FN) | 真陰性 (TN) |

正確接受率 (True Positive Rate , TPR)：在所有實際為正確的樣本中，被正確地判斷為正確之比率。

$$TPR = \frac{TP}{TP + FN}$$

錯誤接受率 (False Positive Rate ,FPR)：在所有實際為錯誤的樣本中，被錯誤地判斷為正確之比率。

$$FPR = \frac{FP}{FP + TN}$$

在第二步驟中的身體密碼鎖的部分，做了 50 組正確密碼鎖和 50 組錯誤密碼鎖的實驗，此系統之辨識密碼鎖的結果如下表 4-2 所示，而其中正確接受次數和正確拒絕次數的比值越高則代表辨識錯誤的情況越低，也較不容易出錯。

在 50 次的測試當中，試著辨識 50 個不同使用者的骨架，由 50 個不同的使用者擺出所設定之動作，將辨識結果記錄下來，表 4-2 左半部顯示：真正擺出正確密碼鎖動作，且系統辨識為正確(即 TP)的次數為 42 次，而真正擺出正確密碼

鎖動作，但系統辨識為錯誤(即 FN)的次數為 8 次；右半部顯示：並未擺出真正密碼鎖動作，但系統辨識為正確(即 FP)的次數為 10 次，而擺出錯誤密碼鎖動作，且系統辨識為錯誤(即 TN)的次數為 40 次。

表4-2 辨識密碼鎖之可能表列

| 密碼鎖辨識 | |
|----------|-------|
| TP=42 | FP=10 |
| FN=8 | TN=40 |
| TPR=0.84 | |
| FPR=0.2 | |

對於手部指令的部分，分別做了 50 組正確指令與 50 個錯誤指令的實驗，此系統之辨識指令的結果如下表 4-3 所示，左半部顯示：正確的比出已設定的手勢，且系統執行正確的指令(即 TP)的次數為 46 次，而正確比出手勢，系統卻執行錯誤或不執行指令(即 FN)的次數為 4 次；右半部顯示：比出錯誤或類似的手勢，系統卻執行正確指令(即 FP)的次數為 2 次，而比出錯誤或類似手勢，系統也判定為錯誤(即 TN)的次數為 48 次。

表4-3 辨識指令之可能表列

| 指令辨識 | |
|----------|-------|
| TP=46 | FP=2 |
| FN=4 | TN=48 |
| TPR=0.92 | |
| FPR=0.04 | |

會造成系統辨識錯誤之偽陽性(FP)與偽陰性(FN)的原因大多都是因為角度的關係，在第二步驟之骨架密碼鎖的部分，若使用者擺出與密碼鎖動作差異不大

的動作時，系統可能會誤判；第三步驟之手勢互動指令的部分，若使用者比出與設定指令相似的手勢，也可能造成系統誤判。

整體來看，此系統的準確率約為七成七，由於此系統共有三個步驟的驗證，主要是希望能將本系統應用在不便於使用滑鼠及鍵盤的場所，可透過使用者本身的操作方式，來降低其系統所產生執行指令時的錯誤發生，若分別來討論，其準確率可提升至九成。其改善的操作方式說明如下：正如系統流程圖，若允許的使用者不小心比出錯誤的手勢指令時，可先比出勾勾指令，即取消目前所選擇的指令，待系統重新進行辨識後，再執行正確指令，如此一來便沒有錯誤的情況產生；若非系統允許的使用者來操作此系統，即使在某一階段出現辨識錯誤的情況，也可以在下一階段被拒絕使用。

第五章 結論與未來展望

在本篇論文中利用 Kinect 的特性，設計出一套高效率且低成本的身分辨識的驗證。首先，使用 Kinect 官方提供的 SDK 來截取人體的三維資訊以進行辨識，並結合設定之指令做應用。先定義使用者的臉部特徵，以此做為第一步驟的辨識，接著定義使用者欲設定之動作來做為第二驗證步驟的辨識身分功能，最後使用從 SDK 中得到的骨架資訊延伸出手部姿勢的偵測，利用手指的辨識來執行所設定之指令，進而結合在應用工具上。由於人體的三維資訊無法輕易的被複製使用，藉著此項優勢，可保障系統的安全性。

辨識系統在未來的工作為如何提高姿勢的多樣性及準確度，另外也可以利用 Kinect 的麥克風來收聲音，並做辨識及判斷來提高系統的安全性。本論文系統只是初步的雛形，將辨識系統視為一個核心，希望在未來研究中可以應用在更多不同領域上，將分述如下：

1. 醫院：將系統架設在特殊檢查機器操作台前，當病人需做檢查時，醫務人員便可陪伴在病人旁邊直接來操控機器，不僅可以達到安撫病人的成效，也可以隨時發現檢查過程中所出現的突發情況，對機器做出即時的修正。
2. 工廠：將系統架設在工廠器具操作台前，當使用者進入系統可偵測的範圍內，即可開始辨識其身分。由於進入工廠皆需身著安全服裝，如安全帽、防塵衣、手套...等，若要利用鍵盤或滑鼠來操作機器，較為不便，經過此系統在先前的身分驗證步驟中，可確保操作員的身分，而不會被輕易的竄改運作內容，在手指偵測所設定的指令中，即便是穿著安全服裝也可以直接來操作機器。
3. 健身房：將系統架設在健身房中不同健身器材上，或是不同的教室門口前，基於使用者付費的概念，假設使用者欲進入瑜珈教室進行課程，經過此系統所提供的辨識及驗證身分功能，便可即時扣款且只收取所使用項目的費用。

參考文獻

- [1] Jan Smisek, Michal Jancosek, and Tomas Pajdla, "3D with Kinect," IEEE International Conference on Computer Vision Workshops (ICCV Workshops), pp. 1154-1160, 2011.
- [2] Tanner Bryce Blair, and Chad Eric Davis, "Innovate Engineering Outreach: A Special Application of the Xbox 360 Kinect Sensor," IEEE Conference on Frontiers in Education, pp. 1279-1283, 2013.
- [3] Jungong Han, Ling Shao, Dong Xu, and Jamie Shotton, "Enhanced Computer Vision with Microsoft Kinect Sensor: A Review," IEEE Transactions on Cybernetics, Vol. 43, Issue 5, pp. 1318-1334, 2013.
- [4] Shujian Han, Shugui Liu, Lishuan Ji, Shaohui Li, Xiaojie Zhang, and Hongwei Zhang, "3D Shape Measurement of Specular Surface Based on Structured Light Coding Method," International Conference on Optoelectronics and Microelectronics (ICOM), pp.565-569, 2012.
- [5] Chadi Albitar, Pierre Graebbling, and Christophe Doignon, "Robust Structured Light Coding for 3D Reconstruction," IEEE 11th International Conference on Computer Vision (ICCV), pp. 1-6, 2007.
- [6] Nicholas M. DiFilippo, and Musa K. Jouaneh, "Characterization of Different Microsoft Kinect Sensor Models," IEEE on Sensors Journal, Vol. PP, Issue 99, pp. 1, 2015.
- [7] Alparslan Yildiz, Abdullah Akay, and Yusuf Sinan Akgul, "Wii Remote Calibration Using the Sensor Bar," IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 7-12, 2011.
- [8] Yihua Lou, Wenjun Wu, Hui Zhang, Haikuo Zhang, and Yongquan Chen, "A

- Multi-User Interaction System Based on Kinect and Wii Remote," IEEE International Conference on Multimedia and Expo Workshops (ICMEW), , pp. 667, 2012.
- [9] Ling Shao, Jungong Han, Dong Xu, and Jamie Shotton, "Computer Vision for RGB-D Sensors: Kinect and Its Applications," IEEE Transactions on Cybernetics, Vol. 43, Issue 5, pp. 131-1317, 2013.
- [10] Zhou Ren, Junsong Yuan, Jingjing Meng, and Zhengyou Zhang, "Robust Part-Based Hand Gesture Recognition Using Kinect Sensor," IEEE Transactions on Multimedia, Vol. 15, Issue 5, pp. 1110-1120, 2013.
- [11] Chih-Hsiao Tsai, and Jung-Chuan Yen, "The Development and Evaluation of A Kinect Sensor Assisted Learning System on the Spatial Visualization Skills," Procedia-Social and Behavioral Sciences Conference, Vol. 103, pp. 991-998, 2013.
- [12] Yuki Oka, and Takeshi Shakunaga, "Real-time Face Tracking and Recognition by Sparse Eigentracker with Associative Mapping to 3D Shape," Image and Vision Computing Conference, Vol. 30, Issue 3, pp. 147-158, 2012.
- [13] Nikolai Smolyanskiy, Christian Huitema, Lin Liang, and Sean Eron Anderson, "Real-time 3D Face Tracking Based on Active Appearance Model Constrained by Depth Data," Image and Vision Computing Conference, Vol. 32, Issue 11, pp. 860-869, 2014.
- [14] Akshay Asthana, Jason Saragih, Michael Wagner, and Roland Goecke, "Evaluating AAM Fitting Methods for Facial Expression Recognition," International Conference on Affective Computing and Intelligent Interaction and Workshops, pp. 1-8, 2009.
- [15] Chien-Cheng Li, "Fast Human Posture Recognition by Heuristic Rules," NTU Institutional Repository, 2006.

- [16] Chien-Cheng Li, and Yung-Yaw Chen, "Human Posture Recognition by Simple Rules," *Systems, IEEE International on Conference Man and Cybernetics*, Vol. 4, pp. 3237-3240, 2006.
- [17] Castiello, C., D'Orazio, T., Fanelli, A.M., Spagnolo, P., and Torsello, M.A., "A Model-free Approach for Posture Classification," *IEEE Conference on Advanced Video and Signal Based Surveillance*, pp. 276-281, 2005.
- [18] Scott Schaefer, and Can Yuksel, "Example-Based Skeleton Extraction," *Proceedings of the fifth Eurographics Symposium on Geometry Processing*, 2007.
- [19] Chih-Chiang Chen, Jun-Wei Hsieh, Yung-Tai Hsu, and Chuan-Yu Huang, "Segmentation of Human Body Parts Using Deformable Triangulation," *International Conference on Pattern Recognition*, Vol. 1, pp. 355-358, 2006.
- [20] Chi-Hung Chuang, Jun-Wei Hsieh, Luo-Wei Tsai, and Kuo-Chin Fan, "Human Action Recognition Using Star Templates and Delaunay Triangulation," *International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, pp. 179-182, 2008.
- [21] Fernando Cassola, Leonel Morgado, Fausto de Carvalho, Hugo Paredes, Benjamim Fonseca, and Paulo Martins, "Online-Gym: A 3D Virtual Gymnasium Using Kinect Interaction," *Research conference on virtual worlds – Learning with simulations*, *Procedia Technology*, Vol. 13, pp. 130-138, 2014.
- [22] Erik Ramos, Moises Ramirez, Eduardo Nila, Diego Figueroa, Jorge Hernandez, Monica Garcia, and Esperanza Perez, "Based Kinect Application to Promote Mixtec Culture," *3rd Iberoamerican Conference on Electronics Engineering and Computer Science*, Vol. 7, pp. 344-351, 2013.
- [23] Rodrigo Ibanez, Alvaro Soria, Alfredo Teyseyre, and Marcelo Campo, "Easy Gesture Recognition for Kinect," *Journal of Advances in Engineering Software*,

Vol. 76, pp. 171-180, 2014.

[24] Kwoon Lee, and Kyungha Min, "An Interactive Image Clipping System Using Hand Motion Recognition," Information Systems, Vol. 48, pp. 296-300, 2015.

[25] Riyad A. El-Iaithy, Jidong Huang, and Michael Yeh, "Study on the Use of Microsoft Kinect for Robotics Applications," IEEE on Position Location and Navigation Symposium, pp. 1280-1288, 2012.

[26] Andrew J. Rettig, Sumit Khanna, Dan Heintzelman, and Richard A. Beck, "An Open Source Software Approach to Geospatial Sensor Network Standardization for Urban Runoff," Interdisciplinary Journal of Computers, Environment and Urban Systems, Vol. 48, pp. 28-34, 2014.

[27] Kinect for Windows. <http://www.microsoft.com/en-us/kinectforwindows/>

[28] PrimeSense

<https://cg2010studio.wordpress.com/2011/11/03/kinect%E7%B0%A1%E4%BB%8B/>

[29] Heresy Space. OpenNI 簡介

https://kheresy.wordpress.com/2011/01/19/openni_1st/

[30] Intel Realsense

<http://www.intel.com.tw/content/www/tw/zh/architecture-and-technology/realsense-overview.html>