

# 應用三維加速度感測器發展空間資訊瀏覽服務 之混搭系統

## Three-dimensional acceleration sensor space for development of information browsing service mash system

李育儒  
國立台東大學  
資訊工程學系  
li9182720025  
@gmail.com

王聖銘  
國立台東大學  
資訊管理學系  
ryan.nttu  
@nttu.edu.tw

黃琬婷  
國立台東大學  
資訊工程學系  
tomatocat129  
@hotmail.com

黃泊澔  
國立台東大學  
資訊工程學系  
handsomeNG  
@hotmail.com

陳冠宏  
國立台東大學  
資訊工程學系  
suck827  
@hotmail.com

### 摘要

近年來隨著科技的發達，嵌入式設備、電池與通信技術的進步，促成了微型感測器 (Sensor) 的發展，並延伸出各種感測器，讓環境自動提供資訊以達到自動化的服務研究，使生活更加便利。本研究中利用加速度感測器裝置透過物理相關原理，推導出座標轉換的關係式，並測試感測器的定位可行性。本研究以 XML 導入系統中，利用資料混搭 (Mashup) 的概念結合 Web Service 之 Open API 並導入服務導向架構 (SOA) 的概念，設計一個由感測器操控的 3D 互動介面，希望藉此建立一簡單具趣味性之互動式介面，透過無所不在的網路環境，達到數位學習生活的延展擴張。

關鍵詞：Mashup、Ubipuitous Computing、情境感知

In recent years, as technology advances, embedded devices, batteries and communication technology advances, led to the development of micro-sensors, and extended all kinds of sensors, so that the environment automatically provide

information to achieve the automated services research, to make life more convenient. In this study, the use of accelerometer sensor devices related to various physical principles, derived coordinates conversion relationship, and to test the feasibility of positioning sensors. In this study, XML import system, using the concept of data mashups with Web Service's Open API and import the concept of service-oriented architecture to design a sensor controlled by a 3D interactive interface in the hope of establishing a simple and fun-oriented The interactive interface, through a ubiquitous network environment, to extend the life of digital learning to expand.

**Keywords** : Mashup、Ubipuitous Computing、context aware

## 1. 導論

今日的世界電腦科技發達、人們通訊方便，但是許多科技產品的操作介面卻有不夠人性化的問題，許多軟體都擁有強大的功能，可是卻是難以操作。網際網路的方便，電腦計算功能的增強，成本價格降低，人們可以便宜取得功能強的電腦計算機讓以往不易完成的運算都變簡單多了，但又有新問題出現了，如資訊安全部分。今日除了利用 RFID 標籤技術取得相關資訊外，也發展出各種感測器，讓環境自動提供資訊以達到自動化的服務研究，加上近年來的發展無所不在運算 (Ubiquitous Computing)，都對人們的生活型態有許多重大的改變。

## 2. 文獻回顧

於本研究中，運用以下幾種概念與技術來完成空間資訊導覽互動介面。利用 Phidget 三軸加速度感測器來感測使用者情境資訊，並藉由 Mashup 的概念結合 Phidgets API、Tween API 和 Papervision3D API。最後以服務導向架構的概念，提供使用者所需資訊，達成空間導覽互動介面。本研究的主程式是用 ActionScript 的程式碼所撰寫編輯而成的，結合了 Flash 一貫強大的動畫能力，加上淺顯易懂的 GUI 介面開發出友善的使用者介面。以下就針對本研究所應用的相關知識與技術做探討。

### 2.1 Mashup 與 SOA

資訊混搭是一種有關資訊分享和支援情境整合的網路應用方法，藉由透過資料和網路 API 來結合現存的網路資源，以創造新的應用。中心概念就是簡單、可用和容易使用 [10]。和過去傳統的整合方式不同，就像應用、資料和外觀整合等方面。例如，Google Maps JavaScript API 可以放置可點擊的標示，透過點擊標示所彈出的各種

資訊來和使用者互動。目前所開發出來的混搭工具和平台為方便後端使用者整合網路上的動態資料，像是 IBM 的 QEDWiki、Yahoo 的 Pipes、Google Mashup Editor 和 Microsoft 的 Popfy 等都是知名和使用者眾多的混搭平台 [12]。

透過資訊混搭與 SOA 概念與技術的應用，是目前資訊服務界面發展的趨勢 [11]。SOA 基本上是一種架構模型，主要由網站服務技術等標準化元件組成，會根據使用需求者的需求分析，服務層是 SOA 的基礎，可直接被應用調用，從而有效控制系統中與軟體代理交會的人為依賴 [3]。SOA 的主要概念是針對企業需求組合而成的一組軟體元件。組合的元素通常包括軟體元件、服務及流程三個部份。當企業面對外部要求時，流程定義外部要求的處理步驟；服務包括特定步驟的所有程式元件，而元件則負責執行工作的程式。由於 SOA 的興起與 Web Services 相關技術成為標準的潮流密不可分，因此常常將 Web Services 與 SOA 架構的實現畫上等號。然而 SOA 架構的實現不限於 Web Services。自 XML、SOAP、WSDL、UDDI 等 Web Services 的技術出現以來，快速成為各界共奉的技術標準，而由其衍生的各種可能應用，其想像空間更是寬廣無限，因而創造了 SOA 概念 [6]。

### 2.2 情境感知技術的應用與發展

情境感知主要將使用者所需資訊，依使用者當時所處之地理環境透過輔助行動載具或是感應器協助，給予使用者適切的資訊 [4]。情境感知的運算 (context aware computing) 可分為主動情境認知 (active context awareness) 及被動情境認知 (passive context awareness) 兩種。主動情境認知是指當系統接收到了情境因素之後，會根據情境因素的內容去改變系統的行為，來回應週遭環境的變化。而在被動情境認知的定義中，系統則是為

有興趣的使用者呈現新的或是更新過的情境因素資訊，或者是保留接收到的情境因素，等使用者需要時(如完成某指定的動作、使用者到達指定的地點或是指定的時間)，再進行下一步的相關服務動作[8]。

情境感知運用的三種重要內涵：一是微運算(micro-computing)或普及運算(Pervasive Computing)、二是使用者界面設計(User interface design)、三是無所不在通訊網路(Ubiquitous communications networks)[3]。

情境感知的因素主要分為以下四類：

- (1) 計算情境(Computing context)：如網路服務品質、頻寬、通訊花費等。
- (2) 使用者情境(User context)：使用者的位置、記錄檔、喜好設定、鄰近使用者等。
- (3) 時間情境(Time context)：每天、日、週的某一時刻或是季節性情境因素。
- (4) 實體情境(Physical context)：溫度、亮度、聲音大小程度等。[9]

## 2.3 人機互動

ACM SIGCHI 會議於 1992 年對 HCI(Human Computer Interaction)下一定義：人機互動是一種評估、設計與應用的原則，其重點是在於給人類使用者使用的互動電腦系統與其相關現象的研究[7]。人類使用者透過一些方法、對話與行為來使用並與電腦互動[2]。

目前市面上所指的人機介面則多界定在狹義定義上的，指在軟體人性化的操作介面上。一個成功的互動模式是一個人可以直接告訴電腦要作些什麼，而不是用其他方式或透過他人的協助間接達成[3, 7]。在趨勢引領下，人機互動模式不斷更新，語音辨識與合成、手寫體與手勢辨識、虛擬實境等技術，均已成為人與電腦互動的通道；其應用範圍也越來越廣，從傳統電腦到個人數位助理(PDA)[1, 5]，甚至行動電話，都受其

影響，而人機介面設計的優良也對整體系統成效的影響越來越大。一個設計良好的人機介面，不但可以在使用初期減少學習時間、提早發揮系統效益，而在學習階段後，更可提升系統終極的整體績效，減低系統發生錯誤的機率[1]。

人機介面的走向，終將從電腦為中心轉變為以人為中心。在目前人機互動系統中，人被稱為使用者，與機器進行對話時，沒有主動控制系統反應的能力。而在未來系統中，人才是主動的參與者，電腦將對人的各種動作做出反應[1, 6, 7]。

目前逐漸普及的觸碰式操作裝置，就是新一代的人機互動操作裝置，現在這些裝置，也加入了語音控制的部份，在未來，其他感應裝置也會加入在其中。近年來許多研究者紛紛投入實體介面、代理人、自然語音辨識等研究，這些研究的共同點在於使電腦形貌反璞歸真的意圖，讓人機互動變得像人際互動一樣簡單、自然並且原始[2]。隨著人機互動模式的推陳出新，新一代人機互動模式被要求包含虛擬實境、語音辨識與合成、手寫體與手勢辨識等技術，而整合這些來自不同感知通道的資訊，有賴多通道介面的發展。開發多通道介面的目的在充分利用人類多種感覺和運動通道的互補特性，來達到使用者與電腦相互間的溝通，從而增進人機互動中的自然性。

## 2.4 Phidgets 三軸感應加速晶片

Phidgets Accelerometer Sensor 是由 Phidgets 公司發行，此裝置是一個每軸能測量  $\pm 3 G$  重力加速度 ( $\pm 29.4 \text{ m/s}^2$ ) 的三軸加速度計，其可用來測量動態加速度(振動)及靜態加速度(重力加速度或傾斜)，Accelerometer Sensor 應用在 PC 進行簡單操作和控制機器人的設備等，並支援 USB 連接埠，也提供一個簡單易用的應用程式界面(API)。本研究所應用的晶片如圖 1 所示。

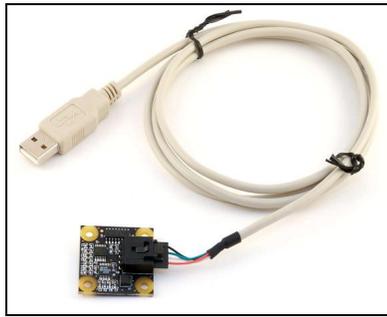


圖 1：Phidgets 三軸加速度感測器

加速規的用途，一般而言，慣性導航系統就是利用加速規提供載具相對於慣性空間的位移變化，是一種測量加速度的裝置。相對於遠距感測的裝置，它測量的是自身的運動。加速規的應用之一是測量重力，特別是使用於重量測定法的加速規上，這樣的裝置被稱為重力計。加速規可能是最簡單的 MEMS 裝置，有時只由一個懸臂和一個重錘組成，利用撓曲和電路來測量加速度。加速規可以測量幾千個 G 的幅度，單軸、二軸、三軸都可以[8]。

### 3. 研究系統架構

本研究使用 Phidget 三軸加速度感測器，Phidget 1059 型硬體設備與電腦連接 Phidgets 應用軟體然後進行數據資料擷取。

我們從運動理論，來了解加速度數據的原理，根據牛頓第二運動定律，加速度是物體速度對時間的變化率，而速度是物體位置對時間的變化率。經由以上我們得知感測器晶片蒐集到的加速度數據必須兩次積分將加速度轉換成位移 (Displacement) 的資訊，並進行相對位置的推導與試驗；因此在本章我們著重於加速度之數據轉換公式推導，再利用 Papervision3D API 設計一個 3D 使用者界面，然後利用感測器晶片操作 3D 界面中的物件，並根據感測器的移動、震動與傾斜進行分析討論。

本研究的實驗介面是架構在 Windows XP 的作業環境上，用 Adobe Flash CS3 撰寫操作介面，並結合 Phidgets 提供的 API 來設計操作介面。硬體架構方面，將加速度感測器以 USB 與電腦作連結，並以此為伺服端，然後透過網際網路取得相關之資訊；軟體架構方面，則以 Flash 之 AVM 做為各軟體間的連結，將 Flash API、Papervision3D API、Tween API 和 Phidgets API 加以整合並應用。圖像背景部份，以 swf 檔案格式藉由 XML 匯入系統中。最後藉由 Phidgets 之驅動程式與伺服器之作業系統相連結，以進行系統程式之運作。主要架構如圖 2 所示。

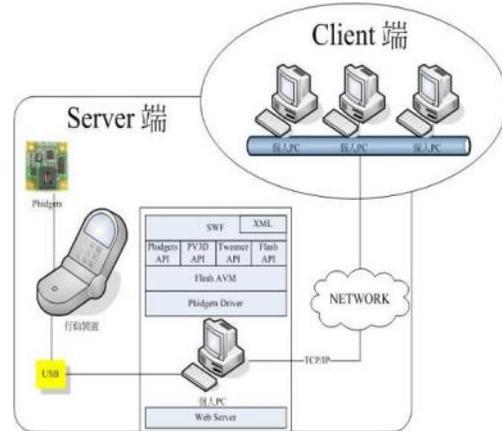


圖 2：軟硬體架構圖

#### 3.1 本研究之硬體設備

Accelerometer Sensor 應用在 PC 進行簡單操作和控制機器人的設備、擷取遊戲機與人機互動的裝置輸入以及藉由結構性震動的測量，偵測出位置或移動等，並支援 USB 連接埠，另外還提供應用程式介面(API)。除此之外 Phidgets 甚至可製作出一台 F15 的模擬飛行器，而且能用來控制所有其他的設備。

Phidget 三軸加速度感測器的數據偵測，在 Phidgets 所提供的 API 中已經設定好 Axis0、Axis1、Axis2 三軸的偵測，且可以自行調整感測器根據某個方向的傾斜來蒐集數據，因此我們在

程式撰寫中繼承 Phidgets API 並把 Axis0 設為 X 來感測左右數據、Axis1 設為 Y 來感測上下數據、Axis2 設為 Z 來感測前後數據，所以依照此設定後，我們就能利用此數據來計算方向。

Phidget 晶片網路應用系統架構如圖 3 所示。感測器運作由裝置的感測、壓力、執行單元擷取數據再透過 USB 與電腦連接並與 Phidgets Library 處理運算，最後可在 Server 端直接輸出資訊也可以 TCP/IP 機制與 Client 連接互動。而其支援 Flash 相關程式的特性，也提供跨平台發展的特性。

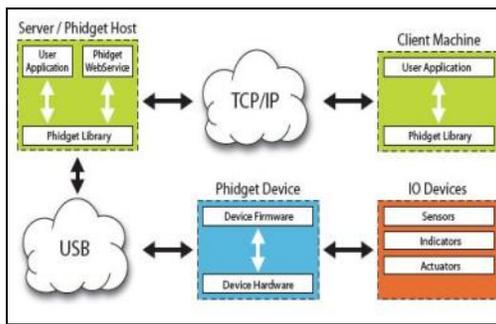


圖 3：Phidgets 晶片網路應用系統架構

### 3.2 加速度資訊之處理

在本實驗中，利用加速度感測器提供的 API 並無法直接取出每個軸向 (XYZ) 的位移量，須先經過數據的處理和演算才能得到位移的資訊，在實驗過程若加速度  $a(t)$ 、速度  $v(t)$  及位移  $d(t)$ ，當感測器為加速度移動時位移的速度及距離公式如下，並根據公式(1)(2)計算出 X、Y、Z 軸向的位移量。

- $v(t)$ ：速度
- $a(t)$ ：加速度
- $d(t)$ ：位移

$$v(T) = v(0) + \int_{t=0}^T a(t) dt \quad (1)$$

$$d(T) = \int_{t=0}^T v(t) dt = v(0) * T + \int_{t=0}^T a(t) dt \quad (2)$$

### 3.3 軟體介面整合

為了提供給使用者一個生動、順暢並且容易操作的軟體介面，本研究結合多種 Open API，如 PV3D、Phidget 以及 Tween API 等。

#### 3.3.1 Papervision3D API

Papervision3D 是一個 Open Source 由 Carlos Ulloa 創始的一款開發 Flash 3D API，簡稱 PV3D(如圖 4)，是一個基於 ActionScript 的開發項目，目的在於實現更加絢麗美觀、功能強大的 Flash 3D Web 應用程式[1]。尤其在 Adobe Flash ActionScript3 發布之後，PV3D 無論在效率、功能上都有很大的提升。且隨著現在 Web2.0、Web-Game 等遊戲的發展，PV3D 的應用也越來越廣並且開放原始碼提供許多在開發 3D 介面的人。在 PV3D 的使用上他主由 viewport、scene、camera、render，四個基本元素來設計 3D 介面。

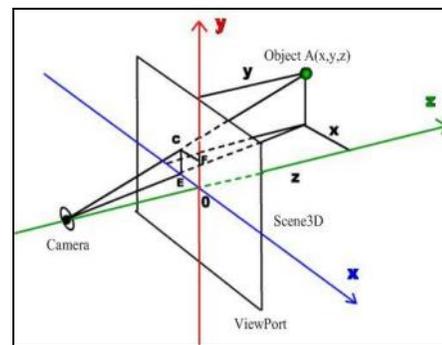


圖 4：PV3D 示意圖

#### 3.3.2 Phidget API

Phidgets API 支援各種程式語言，包括 Java、C++、C#、Flash AS3、Flex 等，主要於開發者可

選擇自己常用或某些系統開發上需要的程式語言，也減少系統開發之困難。在使用 Phidgets API 必須先安裝 Phidgets Library，Phidgets Library 主要功用於 Phidget 感測器與 Phidgets API 的連結，因此在 Phidget 感測器擷取到的數據會透過 Library 應用程式由 Phidgets API 來運算。

### 3.3.3 Tween API

Tween 是一個 Open Source，主要功用是利用 Actionsript 動態的 Tween movieclip 屬性於形狀、顏色、Filter 達到漸變效果(如圖 5)。除此之外 Tween 是根據程式碼的撰寫，並不用時間表以及框架等等的設定，因此在動畫的設計上更易於維護，在動畫基礎上也更容易控制。Tween 提供的 Transition types 適用來定義一個數值，是用一個方法轉變到另一個目標數值，例如選用 Quadratic 或是 Exponentia 的方式轉變，而物件移動或淡出淡入都可以利用 Transition types 建立 easing，因此產生流暢的效果 [2]。

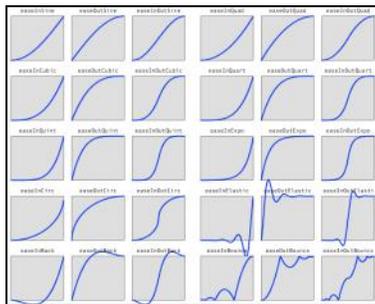


圖 5：Tween 提供的漸變效果

## 4. 實驗結果與分析

本研究所設計的系統功能流程圖如圖 6 所示。為能凸顯本研究為來希能將相關應用嵌入於手機，並發展個人化操作介面的規劃，本研究將 Phidget 三軸加速度感測器直接置入手機殼中，但現階段尚未完成將晶片與程式嵌入於手機的目標。

本研究所應用的介面外觀如圖 7 所示。

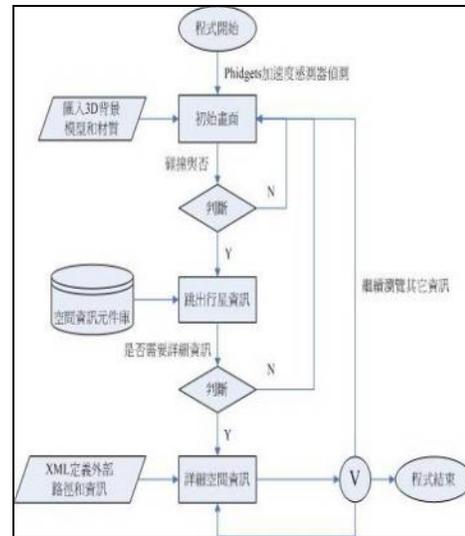


圖 6：系統功能流程圖



圖 7：本研究建構之操作介面圖

於系統起始階段，系統將偵測感測器的連接，並偵測 Phidgets Library 應用程式中有關 Web Service 的功能是否有啟動。接著，進入初始畫面如圖 8 所示。



圖 8：系統操作初始畫面圖

於初始畫面中，首先啟動 Papervision 3D 的 API，載入 3D 背景、行星模型與行星圖片至介面中。另其中以慧星的模型作為介面指標。而隨著操作介面的移動所傳遞的加速度與方向性移動介面中的慧星模型。其操作的介面指標在 Y 軸(上/下)、X 軸(左/右)及 Z 軸(前/後)操作方向性與互動性的結果如圖 9-11 所示。

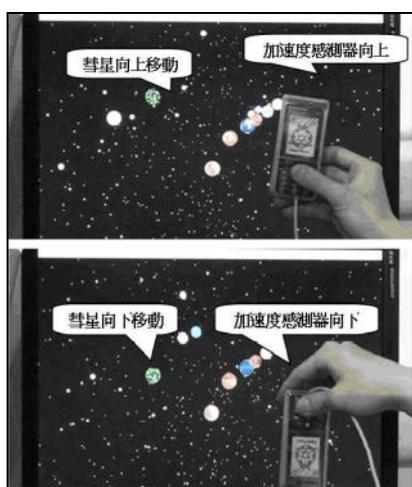


圖 9：系統介面指標 Y 軸(上/下)方向操作示意圖

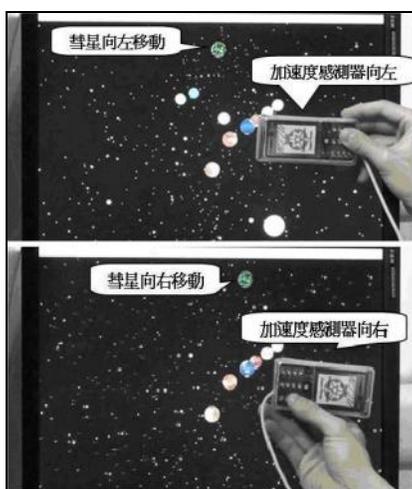


圖 10：系統介面指標 X 軸(左/右)方向操作示意圖

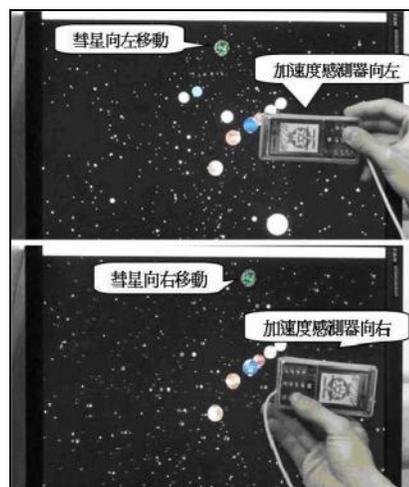


圖 11：系統介面指標 Z 軸(前/後)方向操作示意圖

而透過介面指標的操作，透過加速度與時間，計算介面指標位移量，並於三度空間中找尋並位移至所欲尋找之行星於空間中的位置，介面指標與所選之八大行星，藉由觸碰方式啟動啟動選擇視窗。亦即當慧星碰觸到所選擇的星球時左上角會出現一個資訊視窗，視窗中則出現所選星球的概要介紹。如圖 12 之 A 所示。

而若需進一步的連結相關網頁取得更詳細的資訊，則透過手中的 Phidget 晶片在 X 軸的震動(如圖 12 之 B 所示)，則會啟動啟動 SOA 的服務並透過網路資訊混搭載入所選擇行星的資訊(如 13 之 A 所示)。

而要離開並關閉載入的網頁資訊，並回到起始畫面，則透過手中的 Phidget 晶片在 Z 軸的震動(如圖 13 之 A 所示)，則可將介面回復到起始畫面(如圖 13 之 B 所示)。

## 5. 結論與未來發展

本研究主要在提出如何結合微型感測器與資訊混搭(Mashup)技術，發展互動式三度空間資訊操作介面的結果。透過 Phidget 三軸加速度感測器來感測使用者的情境資訊，並以資訊混搭的技術結合 Flash Actionscripts, Phidgets API、Tween API 和 Papervision 3D API，等開放式應用程式介面的應用，發展一個的互動式三度空間資訊操作介面。此外，藉由服務導向架構概念的應用，本研究也進一步導入其在網路環境中的應用。而研究中藉由加速度晶片的感測，可以讓系統進行的方式從過去的按鍵控制方式，讓使用者有更具互動性與真實性式的操作介面。而在未來發展上，則希能將本研究具跨平台特性的程式碼與裝置，實際嵌入到手機或行動裝置中，一方面發展普適計算的應用，另一方面也藉由資料傳輸與混搭的特性，發展個人化的三度空間操作介面，以及其於未來電子商務，例如：互動式多媒體資訊站(Kiosk)，的應用。而於後續研究建議上，則建議進一步的改進加速度感測器於位移、傾斜資訊的演算，以求更精準的數據以使人機互動的反應更為順暢。另也建議結合更新的網路網頁技術，藉由 SOA 及資訊混搭的導入，發展其多面像的應用。

## 6. 參考文獻

- [1] 吳婷婷、黃國楨、宋天文，「建構情境感知無所不在學習之應用」，TANET 2007，台灣大學主辦，2007。
- [2] 唐國豪，「人機互動：人與機器的對話」，科技發展，18-23頁，2003。
- [3] 張紅，「SOA在高校圖書館資源整合中的應用研究」，圖書館建設，第2007卷，第6期：85-87頁，2007。

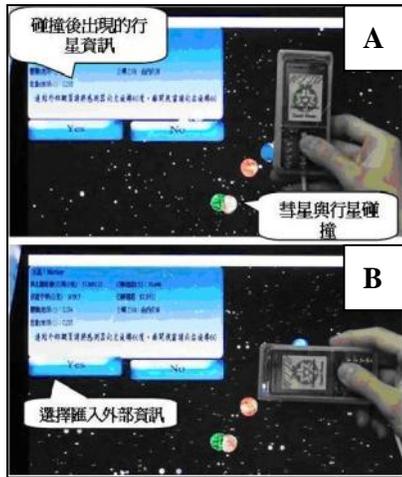


圖 12：介面指標與行星藉由觸碰啟動選擇視窗

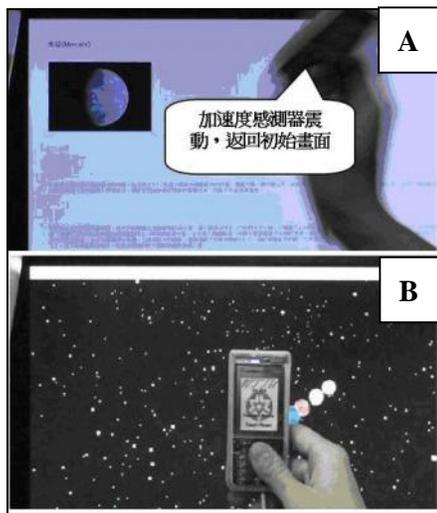


圖 13：介面指標與行星藉由觸碰啟動選擇視窗

上述的實驗結果顯示，本研究所發展的介面，能提供使用者與系統中更有趣且友善的互動性，並能在三度空間的場域中順暢的操作。此外，透過 SOA 的概念與網路資料混搭的應用，能成功的透過網路導入與系統結合的資訊。

- [4] 梁斌，「基於 web2.0 的學習」，中國遠程教育，第003期：39-42頁，2008。
- [5] 陳正忠、石豐銘、蔡明儒、周文陽，「無所不在計算技術在工程上之應用」，中興工程季刊，第98期：9-15頁，2008。
- [6] 劉海珊，「SOA 架構對資訊系統的影響」，iThome 軟體技術應用專刊，2005。
- [7] 蕭顯勝、馮瑞婷，「具情境感知式戶外生態教學系統之規劃與設計」，生活科技教育，第39卷，第5期：28-39頁，2006。
- [8] 魏志瑋、李宗原、林祐民，「以無線感測器網路建立短距離二維剛體移動軌跡即時感知系統」，明道大學資訊工程學系學士學位論文，2008。
- [9] B. N. Schilit and M. M. Theimer, "Disseminating active map information to mobile hosts," *IEEE network*, vol. 8, pp. 22-32, 1994.
- [10] D. Benslimane, et al., "Services Mashups The New Generation of Web Applications," *IEEE Internet Computing*, vol. 12, pp. 13-15, September/October 2008.
- [11] F. Yang, "Enterprise Mashup Composite Service in SOA –User Profile Use Case Realization," presented at the IEEE Congress on Services 2008, 2008.
- [12] J. Yu, et al., "Understanding Mashup Development," *IEEE Internet Computing*, vol. 12, pp. 44-52, September/October 2008.