

## Easy Control

專題編號：104-CSIE-S009  
執行期限：103 年第 1 學期至 104 年第 1 學期  
指導教授：柯開維教授  
專題參與人員：  
101590306 郭鎧璋  
101590347 呂昭陞

### 一、摘要

本專題是以手勢辨識控制為基礎，開發一個能對應各種多媒體、網頁、甚至是遊戲來進行各式各樣的手勢操作，讓使用者無需用鍵盤滑鼠也能操控電腦。

手勢控制方面是以一種叫 Myo 的手環套在手臂上，感測肌肉生理電流變化的方式來判斷各式各樣的手勢。手環感測後會將這些資料用藍芽的方式傳至 windows 系統進行分析判斷手勢，再根據想操控的不同軟體進行手勢對應，也就是說同樣的手勢在不同的軟體能產生不同的效果。

為了能夠讓使用者選擇想要手勢操控的軟體，我們會運用 Windows API 開發一套 Windows Form 讓使用者選擇想要操控的軟體，並讓使用者知道不同手勢所對應的功能。

**關鍵詞：Myo、Windows API (Application Programming Interface)、Windows Form**

### 二、緣由與目的

有時會遇到當下情境不適合操控滑鼠，希望能用簡單的手勢來操控電腦，例如用電腦放著影片，當要暫停、前進、倒退，都需要透過滑鼠或是鍵盤才能完成，這代表需要移動到電腦前才能夠完成需要的功能，我們希望能夠使用簡單的手勢就能達成想要的功能。

而我們找到了一種手環—Myo，他的感測原理讓我們深感興趣，我們覺得這個

手環能夠完成我們的目的一更簡便的操控電腦。

一般大眾所認知的手勢辨識都是以影像辨識的方式來判斷手勢並操控，但 Myo 手環是透過手臂來感測肌肉生理電流的方式來判斷上手勢，且速度比影像辨識快上不少，能夠實現像鍵盤滑鼠那般即時的操作，且不會受到障礙物的阻礙。

為了能操控各式各樣的軟體，我們會用到 windows 視窗去整合各種手勢對應的軟體功能，建立手勢辨識手環與電腦軟體的溝通，完成手勢操控電腦的目的。

### 三、使用技術與方法

#### (一) Windows API

Windows API 是屬於 Windows 作業系統一部分的動態連結程式庫 (DLL)。當您無法自行寫入對等程序時，您就可使用它們來執行工作。例如，Windows 會提供 keybd\_event 的函式，讓程式能夠觸發對應的鍵盤按鍵。

#### (二) Myo SDK

開發 Myo 手環辨識所需的相關 Myo APIs (又稱 Myo Framework APIs 或 Myo SDK APIs)，包含了最基本手勢辨別、肌電訊號大小回傳等。

### 四、研究報告內容

如圖 1 所示，myo 手環會偵測肌肉電流，並將目前的手勢透過藍芽傳到 windows 程式，windows 程式會辨識目前

焦點停駐的視窗，並針對使用者對該視窗自訂的手勢進行判斷，判斷後執行相對應的功能，例如使用者可以針對網頁定義手勢來操控上一頁、下一頁，捲動頁面等功能，亦可針對簡報來定義手勢操控上一頁、下一頁等等。

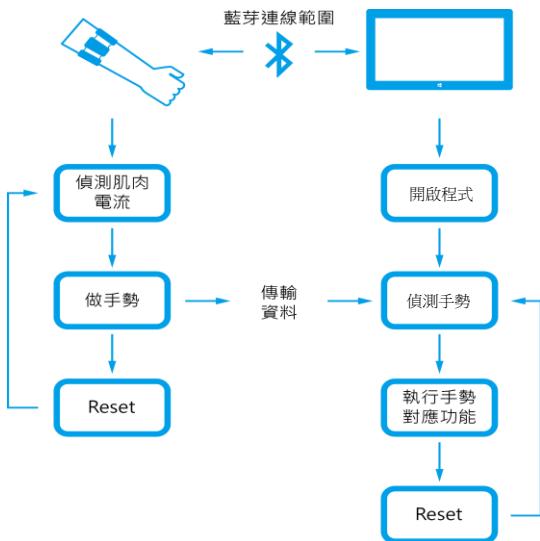


圖 1、系統架構

在程式的撰寫上，由於我們只有一個 myo 手環，因此想出了一個可以讓兩個人共同開發的設計方式，我們將程式分割成前端與後端兩個部分，這兩個程式調用同個 class，如圖 2 所示，使得程式在開發的過程中，後端的程式與前端的程式互不干擾，讓 myo 手環的數量不會造成開發上的限制。

前端的程式(如圖三)可以讓使用者選擇想要操縱的軟體和手勢對應的功能，並將選擇好的手勢資料傳給後端做處理。

後端的程式(如圖四)會接收 myo 傳來的資料，分析 myo 的旋轉方向以及前端傳來的手勢資料，執行相對應的功能。

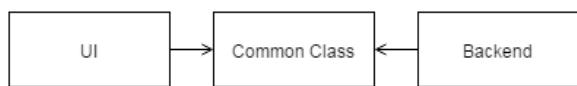


圖 2、程式架構

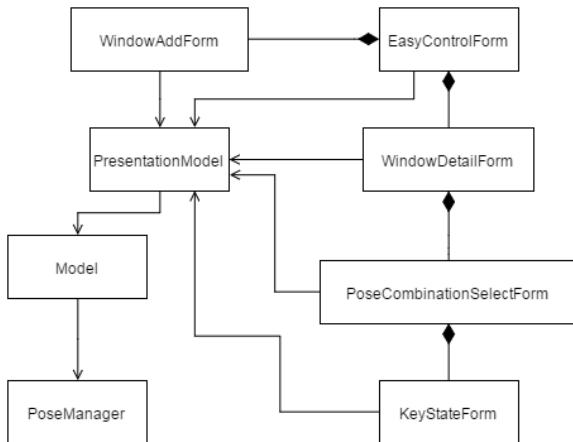


圖 3、前端架構圖

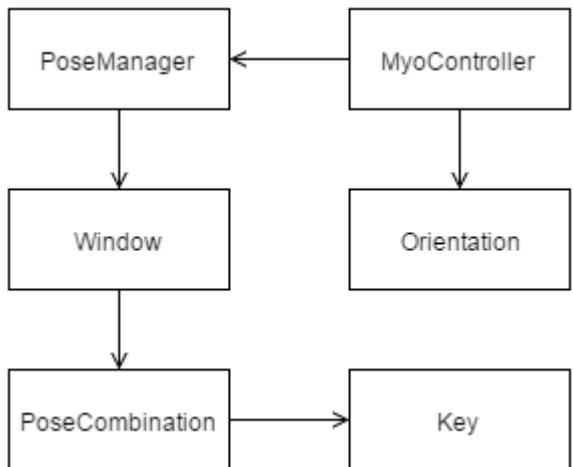


圖 4、後端架構圖

## 參考文獻

- [1] Microsoft Developer Network  
<http://msdn.microsoft.com/zh-TW/>
- [2] Myo SDK 0.8.1: The Myo SDK  
<https://developer.thalmic.com/docs/api-reference/platform/index.html>
- [3] MyoSharp  
<https://github.com/tayfuzun/MyoSharp>