

中華民國第 59 屆全國科學展覽會

作品說明書

科 別： 生活與應用科學(一)(機電與資訊)

組 別：國中組

作品名稱： 天龍八「步」－ 利用足底壓力，製作偵測內八或外八姿勢的穿戴式裝置

關 鍵 詞： Arduino、穿戴式裝置、內八及外八姿勢矯正

編 號：

摘要

已知內八及外八姿勢會影響健康，醫療上利用矯正鞋墊來改善，但大多半年追蹤一次，無法達到即時矯正的效果，因此引發我們的研究動機，我們利用足底壓力分佈的原理，製作能偵測站姿及走路是否內八或外八並具有即時提醒功能的穿戴式裝置，使用方法是當壓力超過設定的門檻值時，手機便會發出圖示及語音警告，我們還建立資料庫進行長期紀錄，期待能更精確地矯正姿勢。我們研發的第一代穿戴式裝置可即時偵測內八及外八的站姿，並在手機上發出圖示及警告，但是速度慢且無法長期追蹤記錄，因此我們進一步研發第二代穿戴式裝置，不只能精準地在站立及走路內外八時，手機發出圖示及語音警告，且速度更快，還可記錄以達到長期追蹤效果。

壹、研究動機

根據 Pollster 波仕特線上市調網的調查結果，「外八」者占整體受訪者的 21.8%，而「內八」者則占整體受訪者的 9.2%（波仕特線上市調網[7]）。大部分內八或外八的人都無法即時察覺自己的姿勢不良，定期就診又無法確實診斷平時的情況。目前主要的矯正方式是先利用儀器量測腳底壓力後，若醫師判定為內八或外八，則給予適當的矯正鞋墊，每隔半年追蹤一次，此方法的缺點是無法讓使用者立即調整姿勢，由於姿勢不良多半會導致許多的傷害，例如內八會影響站姿以及大腦血液循環，外八則容易導致小腿、臀部、腰部及背部的慢性疼痛，所以如果能即時檢測站姿不良的情況，並即時發出警告，對站姿不良的人平時的調整會有很大的幫助。為了方便平時使用，我們想讓 Android 手機，利用藍牙與 Arduino 連結，設計出具提醒功能的穿戴式裝置，希望能夠協助患者即時調整姿勢。

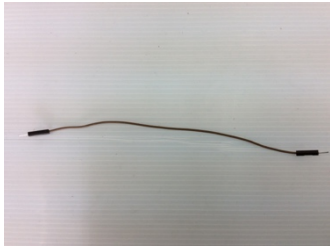
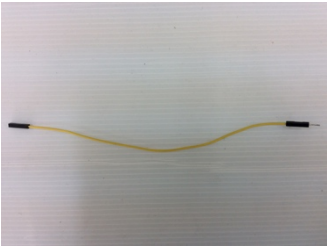

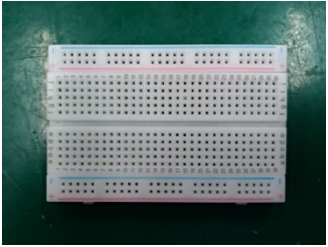


貳、研究目的

- 一、第一代穿戴式裝置:製作偵測內八及外八站姿的穿戴式裝置
- 二、第二代穿戴式裝置:偵測內八及外八姿勢 (包含站姿及走路)的穿戴式裝置

參、研究設備及器材

一、器材

Arduino NANO	HC-06(藍芽)	RFP 307(壓力感測器)
		

杜邦線		
公對公	公對母	母對母
		
麵包板	USB 轉 USB-C 傳輸線	USB 轉 mini USB 傳輸線
		

1000Ω 電阻	焊槍	焊錫
		

剝線鉗	9V 電池
	

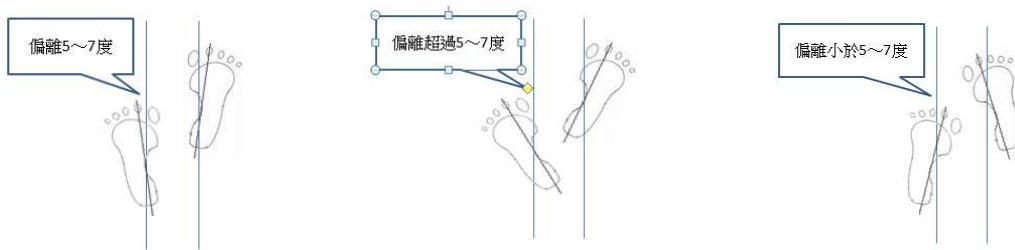
二、程式

Arduino IDE	APP INVENTOR 2
	

肆、研究過程及方法

一、名詞定義及解釋

從我們查詢的資料得知，大部份正常的站姿定義為足的長軸 (即第二腳趾至腳跟) 偏離前方 5~7 度 (健康 2.0[1])，所以在本研究中，我們將小於 5 度定義為內八，大於 7 度定義為外八。另外，內八站立或走路時，雙腳腳底外側壓力會比正常時大；而外八站立或走路時，雙腳腳底內側壓力會比正常時大 (每日頭條[5])(元氣[6])。我們將根據此原理，設計偵測內八或外八的穿戴式裝置。

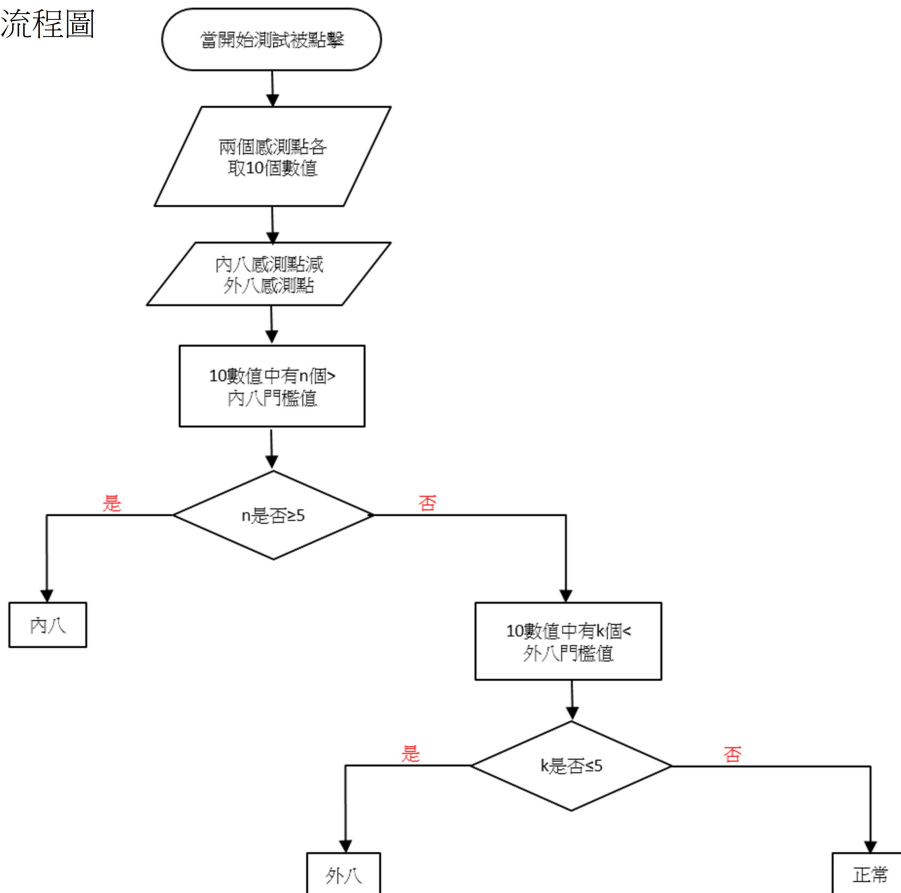


(圖一)不同情況下的站姿，由左到右依序為正常站姿、外八站姿、內八站姿。

二、第一代穿戴式裝置實驗設計

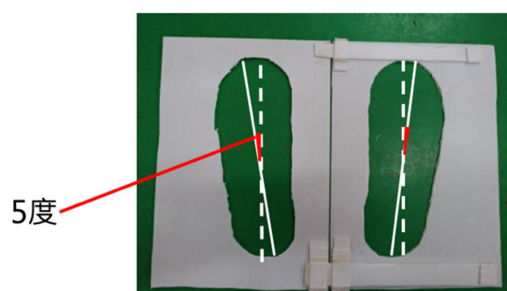
(一) 利用足底雙點壓力差，設計偵測內八或外八站姿的穿戴式裝置

1. 流程圖



2. 正常站姿校正

穿戴裝置後，要先站在校正角度裝置上設定正常站姿的腳底壓力，我們的校正角度裝置如 (圖二)，兩隻腳的長軸角度設計為向外偏移 5 度，使用手機程式時，點擊「開始設定」按鍵後即會開始設定站姿門檻值，一共要設定 5 次 (一次約 1 秒，約共 5 秒)，當設定尚未完成時，手機會顯示「仍需設定__次」，當設定完成時，手機會顯示「開始測試」按鈕。



(圖二)校正角度裝置

3. 站姿門檻值設定

(1) 感測點定位

內八站姿感測點位於足底小指根部由上往下 1 公分，從外側往內 1 公分，外八站姿感測點位於腳跟由下往上 1 公分，從內側往內 1 公分，如(圖三)。我們將站姿壓力感測器貼在鞋墊上的相對位置，如 (圖四)。



(圖三)內八及外八
站姿感測點腳底位置



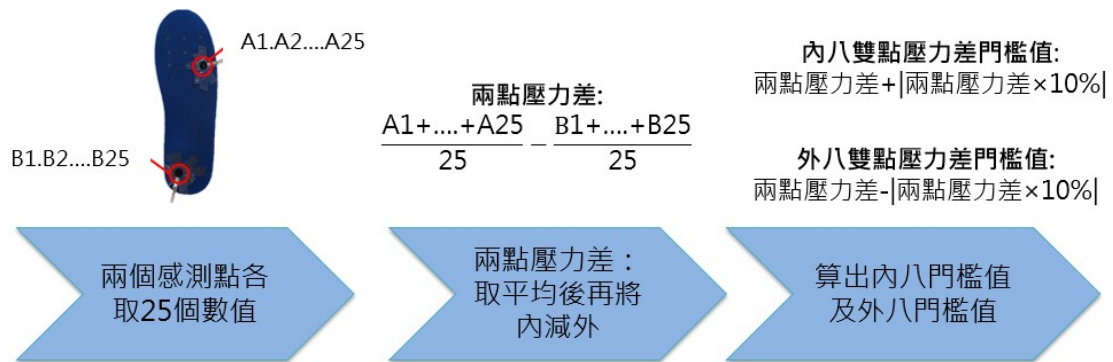
(圖四)內八及外八
站姿感測點鞋墊位置

(2) 雙點壓力差門檻值

在校正的同時，會進行足底雙點壓力差感測，此時，機器總共會在內八站姿感測點 (以下稱內)及外八站姿感測點 (以下稱外)取 25 個壓力數值，並取平均再將內減外，我們將這個數值定義為兩點壓力差，並將兩點壓力差+|兩點壓力差*10%|定義為內八雙點壓力差門檻值；兩點壓力差-|兩點壓力差*10%|定義為外八雙點壓力差門檻值，當設定完成時，手機螢幕會顯示「開始測試」按鈕。

(1) 內八雙點壓力差門檻值 = 兩點壓力差 + |兩點壓力差 * 10%|

(2) 外八雙點壓力差門檻值 = 兩點壓力差 - |兩點壓力差 * 10%|



4. 開始測試

當開始測試時，機器會每輪在內八站姿感測點 (以下稱內) 及外八站姿感測點 (以下稱外) 同時測 10 個壓力值並傳送給手機，由手機進行運算，將內減外 (一輪約 12 秒)，當一輪其中有 5 個數值大於內八雙點壓力差門檻值，手機會判定為內八，並發出圖示及語音警告；若有 5 個數值小於外八雙點壓力差門檻值時，手機就會判定為外八，並發出圖示及語音警告。

(二) 利用足底雙點壓力差探究內八、外八及正常姿勢的準確度

為了進一步探究內八、外八及正常姿勢的準確度，在經過正常站姿校正後，我們將內八及外八雙點壓力差門檻值分別設定為「兩點壓力差」、「兩點壓力差 ± |兩點壓力差 * 5%|」、「兩點壓力差 ± |兩點壓力差 * 10%|」及「兩點壓力差 ± |兩點壓力差 * 20%|」。

在穿戴式裝置上分別表現內八、外八及正常姿勢 3 輪 (30 個數據)，手機記錄所測得的內八、外八及正常次數，然後將測得次數 / 測試次數 * 100%，即為該雙點壓力差門檻值的該姿勢準確度，最後進行比較。

兩點壓力差
 兩點壓力差 ± |兩點壓力差 * 5%|
 兩點壓力差 ± |兩點壓力差 * 10%| (預設)
 兩點壓力差 ± |兩點壓力差 * 20%|

準確度: $\left(\frac{\text{測得次數}}{30}\right) \times 100\%$



三、第二代穿戴式裝置

第一代穿戴式裝置完成，已能成功地偵測是否為內八站姿或外八站姿，但仍有許多可以再精進的地方 (如下表)，經過討論後，我們根據期望的效果，設計第二代穿戴式裝置。

		第一代穿戴式裝置效果	第二代穿戴式裝置期待效果
項目		已達成效果	期待達成效果
姿勢判定	站立	O	O
	走路	X	O
判定速度		慢	快
校正次數		每次穿戴皆須校正	每個人只需校正一次
長期追蹤		X	可長期追蹤紀錄內八外八姿勢的比例
美觀		藍白拖鞋和 Arduino 機器 外接盒	能隱藏 Arduino 機器的鞋子

2. 感測點定位

我們的感測點位置如 (圖七), 內八感測點位於足底小指根部由上往下一公分, 由外側往內一公分, 外八站姿感測點位於腳跟由下往上 1 公分, 往內 1 公分, 外八走路感測點位於足底拇指根部由上往下 1 公分, 從外側往內 1 公分, 我們將壓力感測點設於鞋墊下的相對位置如 (圖八)。



(圖七) 各感測點腳底位置



(圖八) 各感測點鞋墊位置

3. 門檻值設定

(1) 模式判定值

為了要判斷使用者目前是走路還是站立, 所以要站在校正角度裝置上設定模式判定值, 在判定使用者姿勢時會用到此數值。

由於站立時, 腳跟完全著地, 重力向下, 設定模式判定值時, 機器在外八站姿感測點每 0.05 取 1 個壓力數值, 共取 18 個, 約 0.9 秒, 重複以上動作 25 次, 系統會將這 450 個數值取平均定義為模式判定值。

(2) 站姿門檻值

由於內八站姿時, 腳的外側壓力會較正常時大, 而外八站姿時, 腳的內側壓力會較正常時大, 所以要設定站姿門檻值, 在偵測站姿是否內八或外八時會用到此數值。

設定站姿門檻值時, 要先站在我們的校正角度裝置, 機器會在外八站姿感測點及內八感測點每 0.05 秒各取 1 個壓力數值, 同一時間的外八站姿和內八感測點測得的數值稱為 1 組, 共取 18 組, 約 0.9 秒, 系統會將每一組的內八感測點減外八感測點, 重複以上動作 25 次, 並將這 450 個數值取平均定義為站姿兩點差。

(a) 內八站姿門檻值=站姿兩點差+|站姿兩點差*10%|

(b) 外八站姿門檻值=站姿兩點差-|站姿兩點差*10%|

(3) 走路門檻值

由於內八走路時，腳的外側壓力會較正常時大，而外八走路時，腳的內側壓力會較正常時大，所以要設定走路門檻值，在偵測走路是否內八或外八時會用到此數值。

設定走路門檻值時，要先站在我們的校正角度裝置，機器會在外八走路感測點及內八感測點每 0.05 秒各取 1 個壓力數值，同一時間的外八走路和內八感測點測得的數值稱為 1 組，共取 18 組，約 0.9 秒，找出 18 組數值中，2 個數值相加最大的那一組，再將那一組中的外八減內八感測點數值，重複以上動作 25 次，並將這 25 個數值取平均定義為走路兩點差。

(a) 內八走路門檻值=走路兩點差-|走路兩點差*10%|

(b) 外八走路門檻值=走路兩點差+|走路兩點差*10%|

4. 開始測試

開始測試時，機器會在 3 個點每 0.05 秒取 1 個壓力數值，共取 18 個壓力數值，約 0.9 秒，一輪取 3 次，約 2.7 秒，當外八站姿感測點的 18 個壓力數值中有 15 個壓力數值大於設定的模式判定值 (代表腳跟幾乎貼地)，系統就會切換為站姿模式，而當外八站姿感測點的 18 個壓力數值中少於 15 個壓力數值大於設定的模式判定值 (代表腳跟離地)，系統便會切換為走路模式。

在站姿模式時，系統會將內八感測點測得的數值減外八站姿感測點測得的數值，如果相減得到的 18 個數值中超過 15 個數值大於設定的內八站姿門檻值，系統便會記錄為內八站姿 1 次，逐次累加；若小於外八站姿門檻值，系統便會紀錄為外八站姿 1 次，逐次累加，若一輪中有 2 次以上內八，系統便會傳送到手機，發出「內八」的圖示及語音警告；反之，一輪中有 2 次以上外八，系統便會傳送到手機，發出「外八」的圖示及語音警告。

在走路模式時，我們將同一時間的內八和外八感測點測得的數值稱為 1 組，系統將找出 18 組數值中，2 個數值相加最大的那一組 (代表重心在感測點上)，再將那一組中的外八走路感測點減內八感測點數值，如果小於設定的內八走路門檻值，系統便會記錄為內八走路 1 次，逐次累加；若大於外八走路門檻值，系統便會紀錄為外八走路 1 次，逐次累加，若一輪 3 次中有 2 次以上內八，系統便會傳送到手機，發出「內八」的圖示及語音警告；反之，一輪 3 次中有 2 次以上外八，系統便會傳送到手機，發出「外八」的圖示及語音警告。因一般人每步時間約 0.6 秒，所以走 5 步，約 3 秒，即可完成一次判別。

伍、研究結果及討論

一、第一代穿戴式裝置:製作偵測內八及外八站姿的穿戴式裝置

(一) 利用足底雙點壓力差，製作偵測內八或外八站勢的穿戴式裝置

我們將寫完的程式植入 **Arduino** 機器的主機板中，連結內八及外八感測點，貼在鞋墊上，放入鞋子中，再將電線及主機板收入自製 3D 盒子中，固定在鞋子上，為了方便更換電池，用魔鬼氈將電池固定在 3D 盒子上，結果如 (圖九)。

開始使用穿戴式裝置時，要先搜尋並配對 **Arduino** 機器中的藍牙裝置，手機顯示如 (圖十)及 (圖十一)，因為有左右腳 2 個主機板，所以一共要配對 2 個，配對完成後，手機會顯示「開始設定」按鈕 (如圖十二)，點擊後即開始進行正常站姿校正，一共要設定 5 次，設定完成後手機會顯示「開始測試」按鈕 (如圖十三)，點擊後即開始測試，測試時手機螢幕會顯示各個壓力感測點的即時壓力數值，當足的長軸向外偏移 20 度時 (如圖十六)，手機便會發出外八的圖示及語音警告 (如圖十四)；而當足的長軸向內偏移 10 度時 (如圖十七)，手機便會發出內八的圖示及語音警告 (如圖十五)。



(圖九)第一代穿戴式裝置實品

螢幕截圖



(圖十)點擊搜尋藍芽裝置



(圖十一)點擊配對藍芽裝置



(圖十二)站在校正角度裝置上點擊開始設定



(圖十三)點擊開始測試



(圖十四)外八時發出圖示及語音警告



(圖十六)向外偏移 20 度



(圖十五)內八時發出圖示及語音警告

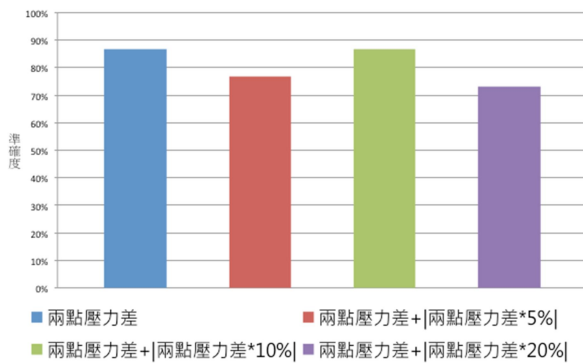


(圖十七)向內偏移 10 度

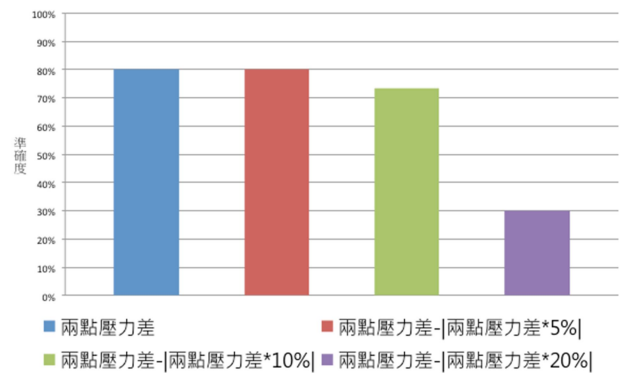
(二) 利用足底雙點壓力差探究準確度

由我們的實驗結果得知，當內八雙點壓力差門檻值為「兩點壓力差」或「兩點壓力差+|兩點壓力差*10%|」時，所測得的內八姿勢準確度最大，都是 87% (如圖十八)；當外八雙點壓力差門檻值為「兩點壓力差」或「兩點壓力差-|兩點壓力差*5%|」時，所測得的外八姿勢準確度最大，都是 80% (如圖十九)；當內八及外八雙點壓力差門檻值為「兩點壓力差±|兩點壓力差*20%|」時所測得的正常姿勢準確度最大，為 90% (如圖二十)。

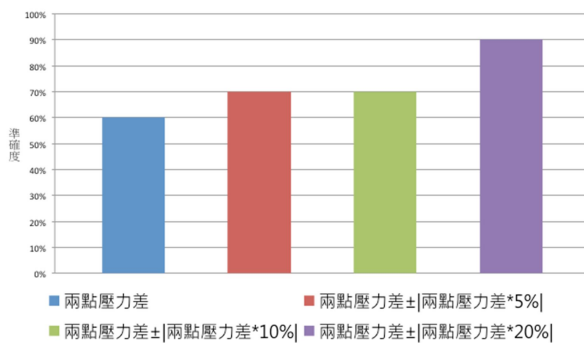
當內八及外八雙點門檻值為「兩點壓力差」時，雖然內八及外八姿勢的準確度較高，但是正常姿勢的準確度較其他低，而當內八及外八雙點門檻值為「兩點壓力差±|兩點壓力差*20%|」時，雖然正常姿勢的準確度較高，但外八姿勢的準確度極低，所以內八及外八雙點門檻值為「兩點壓力差±|兩點壓力差*5%|」或「兩點壓力差±|兩點壓力差*10%|」時，會較為準確。最後折衷將內八及外八雙點門檻值設為「兩點壓力差±|兩點壓力差*10%|」。



(圖十八)不同內八雙點壓力差門檻值偵測內八姿勢的準確度



(圖十九)不同外八雙點壓力差門檻值偵測外八姿勢的準確度



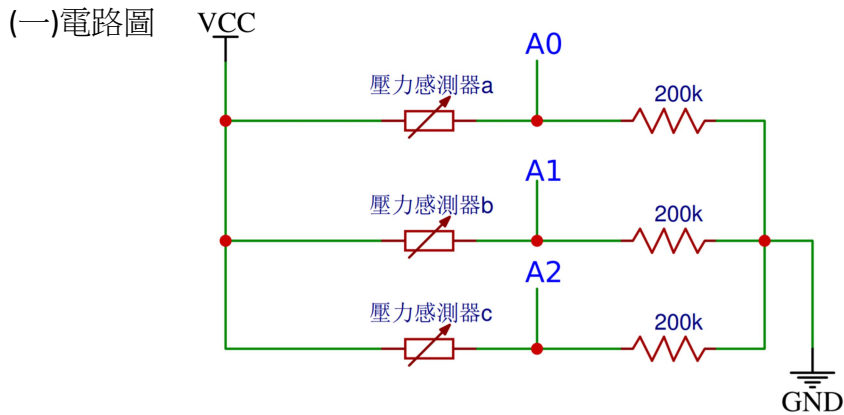
(圖二十)不同內八及外八雙點壓力差門檻值偵測正常姿勢的準確度

(三) 第一代裝置改進方向

第一代穿戴式裝置完成，雖能成功地偵測是否為內八站姿或外八站姿，但仍有許多可以再精進的地方 (如下表)，我們將這些可再精進的地方一一列出並檢討原因，討論如何改進，評估是否可行後，進一步著手進行，根據期望的效果，設計第二代穿戴式裝置。

項目		第一代裝置效果		第二代期望達成效果	
		達成效果	原因	預期達成效果	改進方法
姿勢判定	站立	0	站立時的壓力不會有太大的變化，故運算邏輯較簡單，一開始較容易切入	0	根據走路的步態壓力分佈原理設計偵測走路姿勢的演算法
	走路	X		0	
判定速度		慢 (從開始測定到第一次判定約 14 秒)	傳送數據時，為確保確實傳達，系統與手機端溝通太久	快	調整至必定傳達，不必溝通
校正次數		每次穿戴皆須校正	校正的程式運算在手機中進行，但無記憶功能，故每次皆需校正	每個人只需校正一次	我們先將門檻值設定到的系統裡面
長期追蹤		X	並無設計手機資料庫，故無法長期追蹤	0	我們新增手機資料庫功能，能夠追蹤 30 天數據，方便使用者或其家長檢討站姿改良狀態
美觀		藍白拖鞋和 Arduino 機器外接盒	一開始僅將主機板放置於 3D 列印的盒子內，再黏於鞋底	包覆式球鞋和隱藏式 Arduino 機器	我們將主機板設計於鞋底，外觀幾乎與一般球鞋無異

二、第二代穿戴式裝置: 製作偵測內八及外八姿勢的穿戴式裝置



在我們設計的電路圖中，VCC 代表的是輸入電壓 (5V)，A0、A1、A2 代表的是接收電流，200k 代表的是分壓電阻的電阻值，GND 代表的是接地。

(二) 第二代穿戴式裝置

我們將寫完的程式植入 Arduino 機器的主機板中，連結各個感測點，貼在紙板上後，放入鞋子中，再將鞋墊放入，將主機板固定於鞋底凹槽處，如 (圖二十二)，為了方便更換電池，將 9V 乾電池放置於後腳跟，結果如 (圖二十一)。

開始使用穿戴式裝置時，要先搜尋並配對 Arduino 機器中的藍牙裝置，手機顯示如 (圖二十三)及 (圖二十四)，因為有左右腳 2 個主機板，所以一共要配對 2 個，配對完成後，手機會顯示「開始測試」按鈕 (如圖二十五)，點擊後即開始測試，當足的長軸向外偏移 20 度站立時 (如圖二十八)或走路 (如圖二十九)，系統便會發出外八的圖示及語音警告 (如圖二十六)，而當足的長軸向內偏移 10 度站立時 (如圖三十)或走路 (如圖三十一)，系統便會發出內八的圖示及語音警告 (如圖二十七)。我們 Arduino 程式碼如附錄，而 App Inventor 程式方塊的 pdf 檔如 (圖三十二)



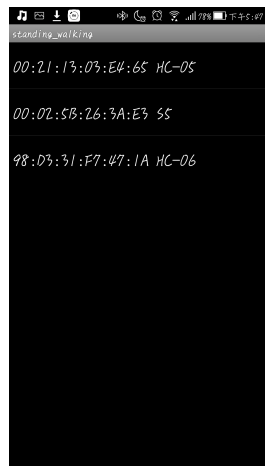
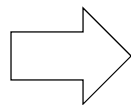
(圖二十一)穿戴式裝置正面照



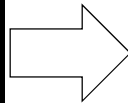
(圖二十二)穿戴式裝置背面照



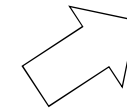
(圖二十三) 點擊
搜尋藍芽裝置



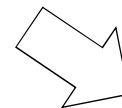
(圖二十四) 點擊
配對藍芽裝置



(圖二十五)
點擊開始測試



(圖二十六) 外八時發
出圖示及語音請告



(圖二十七) 內八時發
出圖示及語音警告



(圖二十八)
向外偏移 20 度站姿



(圖二十九)
向外偏移 20 度走路



(圖三十)
向內偏移 10 度站姿



(圖三十一)
向內偏移 10 度走路



<https://ppt.cc/f1QHQx>

(圖三十二)App Inventor 2 程式方塊 pdf 檔

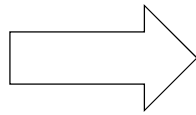
(三) 長期追蹤紀錄

當開始測試時，手機會取得當下時間還有內八及外八姿勢 (走路加站姿) 的次數，並儲存到手機的資料庫裡，一共可以存 30 天的數據，將內八及外八的次數除以測試次數後換算成百分比，接著四捨五入至個位數，當內八或外八比例達「0%~19%」時，手機程式中當天內八或外八儲存格的底色會顯示成白色；「20%~39%」時會顯示成綠色；「40%~59%」時會顯示成黃色；「60%~79%」時會顯示成橘色；「80%~100%」時會顯示成紅色 (如圖三十五)，以達警示效果，手機可以顯示七天的內八及外八比例，能讓受試者簡易地透過觀察儲存格顏色來檢視姿勢情形。

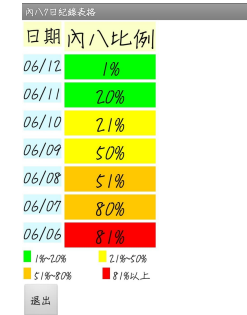
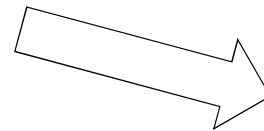
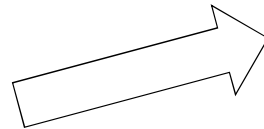
手機會將 30 天內的內八及外八比例繪製成折線圖，因此可以透過觀察折線圖變化來檢視是否有逐漸矯正 (如圖三十六)。



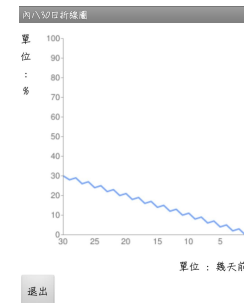
(圖三十三)
點擊記錄按鈕



(圖三十四)
點擊要查看的圖表



(圖三十五)7天記錄表格



(圖三十六)30天追蹤折線圖

(四)第一代及第二代穿戴式裝置效果比較

		第一代穿戴式裝置效果	第二代穿戴式裝置效果
項目		達成效果	達成效果
姿勢判定	站立	0	0
	走路	X	0
判定速度		慢 (從開始測定到第一次判定約 14 秒)	快(從開始測定到第一次判定約 2.7 秒)
校正次數		每次穿戴皆須校正	每個人只需校正一次
長期追蹤		X	1. 可顯示 7 天姿勢狀況 2. 可追蹤 30 天內各姿勢比例
美觀		藍白拖鞋和 Arduino 機器外接盒	包覆式球鞋和隱藏式 Arduino 機器

陸、結論

- 一、我們的第一代穿戴式裝置，受試者經過正常站姿校正，並在表現內八或外八站姿時，手機端都會確實發出圖示及語音警告。且在門檻值為「兩點壓力差 \pm |兩點壓力差*10%|」時，裝置能達到最好的準確度。
- 二、我們的第二穿戴式裝置，能分辨受試者表現站姿或走路，並在表現內八及外八的站姿或走路時，手機端都會確實發出圖示及語音警告，另外，受試者還能夠在手機中查看過去 7 天的內外八比例表，並看到 30 天的折線變化圖，協助追蹤並矯正姿勢。

柒、未來展望

我們希望未來能夠進一步探究以下幾點

- 一、給予壓力感測器不同重量，探討重量對 **Arduino** 機器測得數值的關係並繪出折線圖。
- 二、利用我們研發的第二代穿戴式裝置，蒐集大量正常走路及站姿的人，藉由腳底的感測器測得的數值，得到平均值，期待達到一般化設定。
- 三、探究第二代穿戴式裝置在走路時不同門檻值的準確度。
- 四、找到內八及外八受試者進行測試，並評估我們穿戴式裝置矯正的效果。

捌、參考資料

(一)、健康 2.0(民 107 年 4 月)。走路外八不只是美觀問題，還害你「從腰背痛道小腿」[新聞]。取自 <https://health.tvbs.com.tw/medical/308720>

(二)、每日頭條(民 105 年 12 月)。「內八字」走路會傷害大腦？我勸你趕緊改過來[網路新聞]。取自 <https://kknews.cc/zh-tw/health/mgv6ogz.html>

(三)、天才領袖-王宏哲(民 104 年 9 月)。常見兒童發展問題-走路內八的改善方法[部落格文章資料]。取自

<https://www.leaderkid.com.tw/2015/09/25/%E5%B8%B8%E8%A6%8B%E5%85%92%E7%AB%A5%E7%99%BC%E5%B1%95%E5%95%8F%E9%A1%8C-%E8%B5%B0%E8%B7%AF%E5%85%A7%E5%85%AB%E7%9A%84%E6%94%B9%E5%96%84%E6%96%B9%E6%B3%95/>

(四)、力邁物理治療所。內八及外八異常步態[部落格文章資料]。

取自 <http://painless.idv.tw/foot/index4.htm>

(五)、每日頭條(民 106 年 3 月)。內八字腳對跑步有多大影響[新聞]。

取自

https://kknews.cc/health/z65qvxp_amp?fbclid=IwAR0O8LVj315laq4AruzzQYOxgHKbQmUEYr8oO2a3elwts6Mmsgcydt9CBwE

(六)、元氣網。走路內外八 加速膝關節退化[新聞]。

取自

<https://health.udn.com/health/amp/story/5967/2639052?fbclid=IwAR3xZDBD3zDzzQP-vlRdxOrSi8OuwsanUsEDSQDtjFXbse05aZviZKzLhl>

(七)、Pollster 波仕特線上市調網(民 105 年 3 月 17 日)。走路「內八」或「外八」是否有矯正的必要？[新聞]。取自

<https://n.yam.com/Article/20160517705688?fbclid=IwAR0yJnEEkK3Pn33OMWwjTzepLwc5msPEkcgW89Op9kQjYKQ8Gk8B-WhNN9A>

附錄

Arduino IDE 程式

```
#include <SoftwareSerial.h> //引用函式庫
#include <Wire.h> //引用函式庫
#define fsr0_pin A0 //定義內八感測點腳位
#define fsr1_pin A1 //定義外八走路感測點腳位
#define fsr2_pin A2 //定義外八站立感測點腳位
int mode = 0; //用以確認模式(站立 or 步行)
int in_times = 0; //用以計算內八次數
int out_times = 0; //用以計算外八次數
int stand_times = 0; //用以計算腳跟著地比例
int test_times = 0; //用以計算測試組數
int stand_in_times = 0; //用以計算站立內八次數
int stand_out_times = 0; //用以計算站立外八次數
int judge_mode_standard = 500; //定義模式判定值
int walking_in_standard = -300; //定義走路內八門檻值
int walking_out_standard = 300; //定義走路外八門檻值
int standing_in_standard = -300; //定義站立內八門檻值
int standing_out_standard = 300; //定義站立外八門檻值
int A[20] = {0}; //走路外八感測點測得數值(20 個)
int B[20] = {0}; //內八感測點測得數值(20 個)
int C[20] = {0}; //站立外八感測點測得數值(20 個)
SoftwareSerial BTSerial(5, 6); //定義藍芽腳位
void send() //傳輸走路 or 站立姿勢至手機之函式
{
  if ( test_times == 3) //當測試 3 次時
  {
    byte situation [1]; //建立一個準備傳送的封包
    if ( in_times > 2 ) //當 3 次中有 2 次測得內八
    {
      situation[0] = 100; //手機端設定收到 100 判斷為內八
    }
    else if ( out_times > 2 ) //當 3 次中有 2 次測得外八
    {
      situation[0] = 101; //手機端設定收到 101 判斷為外八
    }
    else
    {
      situation[0] = 102; //手機端設定收到 102 判斷為正常
    }
    in_times = 0; //重新計算 ↓
    out_times = 0;
    stand_in_times = 0;
    stand_out_times = 0; //重新計算 ↑
    BTSerial.write(situation[0]); //傳出封包
  }
}
void test() //讀取感測點壓力數值之函式
{
  for ( int a = 0 ; a < 18 ; a++) //重複執行 18 次
  {
    A[a] = analogRead(fsr0_pin);
    B[a] = analogRead(fsr1_pin);
    C[a] = analogRead(fsr2_pin);
    if (C[a] > judge_mode_standard) //若腳跟感測點數值大於模式判定值
    {
      stand_times++; //腳跟著地次數+1
      delay(50); //每 0.05 秒執行一輪
    }
  }
}
void setup()
{
  BTSerial.begin(9600); //設定藍芽速率
  Serial.begin(9600); //設定序列埠速率
}
void loop()
{
  if ( BTSerial.available() > 0 ) //檢查藍芽是否有資料可供讀取
  {
    switch (BTSerial.read()) //藍芽讀取到的資料
    {
      case 'a': //當藍芽讀取到的資料為 a(開始測試鈕被按下
```

```

{
test(); //執行 test 函式
test_times++; //測試次數+1
if ( mode == 0 ) //mode0 代表走路模式
{
int b = A[0] + B[0]; //找出外八走路感測點+內八感測點最大的一組↓
int team = 0;
for ( int c = 0; c < 18 ; c++ )
{
if ( A[c] + B[c] > b )
{
b = A[c] + B[c];
team = c;
}
} //找出外八走路感測點+內八感測點最大的一組↑
if ( A[team] - B[team] > walking_out_standard ) //利用和最大之組別的走路兩點壓力差判斷是否為外八
{
out_times = out_times + 1;
}
else if ( A[team] - B[team] < walking_in_standard ) //利用和最大之組別的走路兩點壓力差判斷是否為內八
{
in_times = in_times + 1;
}
}
else //非走路模式→站立模式
{
for ( int d = 0 ; d < 18 ; d++ ) //重複執行 18 次
{
if ( B[d] - C[d] > standing_in_standard ) //利用每一組的站立兩點壓力差判斷是否為內八
{
stand_in_times = stand_in_times + 1;
}
else if ( B[d] - C[d] < standing_out_standard ) //利用每一組的站立兩點壓力差判斷是否為內八
{
stand_out_times = stand_out_times + 1;
}
}
if ( stand_in_times > 15 ) //若 18 組數值中超過 15 組測得內八，則將內八次數加一
{
in_times = in_times + 1;
stand_in_times = 0;
}
else if ( stand_out_times > 15 ) //若 18 組數值中超過 15 組測得外八，則將外八次數加一
{
out_times = out_times + 1;
stand_out_times = 0;
}
}
}
send(); //執行 send 函式
if ( test_times == 3 ) //完成一輪測試時(測試次數達 3 次時)
{
if ( stand_times > 12 ) //若腳跟著地比例大於 12
{
mode = 1; //設定模式為站立
}
else
{
mode = 0; //設定模式為走路
}
}
stand_times = 0; //重新計算內八次數
test_times = 0; //重新計算測試次數
}
while ( BTSerial.available() == 0 ); //當藍芽有資料可供讀取時，結束測試
break; //跳出迴圈
case 'b':
{
in_times = 0; //當藍芽讀取到的資料為 b(停止測試鈕被按下)
out_times = 0; //重新計算內八次數
stand_times = 0; //重新計算外八次數
test_times = 0; //重新計算腳跟著地比例
stand_in_times = 0; //重新計算測試次數
stand_out_times = 0; //重新計算站立內八次數
break; //重新計算站立外八次數
}
}
}

```