

腳底壓力辨識結合機器學習之辨識效果分析與研究

指導教授：林均翰教授

謝尚恆 撰

摘要

近年來，生物辨識技術的興起，讓身份的認證方式不再和以前一樣，只能局限於個人的帳號密碼，像是指紋辨識、虹膜辨識等相關技術也已經普及，不僅使生活更為便利，這些技術也讓身份的認證變得更安全，也更為可靠。

不過在廣大的生物辨識市場之中，系統成本與辨識度的考量要如何達成平衡，一直都是辨識系統難以普及化的重點議題之一。在過去研究發現，系統在特徵提取的結果與系統著重於機器學習效果的比例較少，在訓練時所耗費的成本也較無研究。

所以本專題主要在於研究探討分析腳底壓力的資訊，從資料集取出特徵，並與機器學習搭配組合，將機器學習運用在辨識資料，藉此創造出快速取得腳底壓力資訊，且快速訓練，並擁有高準確率的系統模組，接著並進一步根據系統辨識率與感測器感測狀況，調整感測器數量來達到節省成本的目的。

目錄

目錄.....	3
第一章 研究動機與背景.....	4
第二章 相關文獻及既有系統探討.....	5
第一節 地毯型感測系統.....	5
第二節 鞋墊型感測系統.....	5
第三節 現況探討.....	7
第三章 研究方法及步驟.....	8
第一節 開發工具與軟硬體需求.....	8
第二節 系統結構與操作流程.....	8
第三節 機器學習.....	9
3.3.1 SVM(Support Vector Machine)	9
3.3.2 RF(Random Forest)	9
3.3.3 GB(Gradient Boosting)	10
3.3.4 kNN(k-Nearest Neighbors)	10
3.3.5 XGBoost.....	10
第四章 效能評估.....	11
第一節 設計實驗與特徵設定.....	11
4.1.1 長寬.....	11
4.1.2 重心.....	12
4.1.3 訓練樣本設定與準確度判定標準.....	13
第二節 演算法對資料的辨識準確度與訓練時間.....	13
4.2.1 平均資料.....	13
4.2.2 原始資料.....	15
第三節 特徵與辨識準確度的關係.....	16
第五章 結論.....	17
第一節 各組員的貢獻.....	17
第二節 後續研究方向.....	17
參考文獻.....	18

第一章 研究動機及背景

現代人使用網路和資訊進行交換、交流已經和以前不同，不再是只能靠手機、電腦，更不是郵差送信或飛鴿傳書，在現代像是智慧眼鏡、智慧手錶等各式各樣的穿戴裝置也能無時無刻地進行訊息的交換、流通，這時候安全性就會變得很重要，因為資訊一不小心就會四處流通。現在生物的識別技術已普及於日常生活，3C 用品支援人臉辨識、虹膜辨識、指紋辨識等技術已經不足為奇，還有使用腳底壓力進行辨識的技術，這項技術也有許多論文研究過，利用腳底壓力識別的好處，就是它沒有侵犯性也難以偽裝，加上每個人的腳底壓力數據都不太一樣，使得腳底壓力是個不錯的辨識特徵，在許多領域都有應用價值。

其中感測腳底的方法大致上分為兩種，分別是使用地毯、鞋墊去感測受試者的腳底壓力數據，然後將測得的數據使用機器學習進行資料訓練並辨識，以分辨出不同的人。但是在現實情況下，這樣的方法成本高，測量數據上也不方便，而且機器學習與辨識的時間長，使得使用者的體驗不好，進而讓使用腳底壓力識別的意願減少。

因此，我想要藉由這份專題提出一套快速收集資料、低成本的腳底壓力辨識系統，透過收集腳底的壓力資訊收集，提取為腳底特徵，配合近年來較為新穎的機器學習，使系統辨識率提高，最後達成高辨識率、低成本、方便收集與訓練的一套腳底壓力辨識系統。

這份專題會以達到又快速、又準確的辨識系統為核心。第二章將會對現有相關文獻做研究與探討，從中可得出現有文獻的缺點與不足。在第三章我們會先提出整個系統的流程，提出我們設計的硬體，以及使系統更加快速與方便的方法。第四章則是對我們所提出的方法進行實驗，評估其效能與成效，並從中找出最佳的機器學習模型，也就是訓練速度很快，又能達到很高的辨識準確度。最後，第五章會提出此套腳底壓力辨識系統的結論與未來展望。

第二章 相關文獻及既有系統探討

目前的文獻內容中，常見的腳底感測方法有兩種，一種是將感測器放置於長方形的地毯下面，受試者只要穿鞋子走過後，感測器就能取得腳底的資訊，另一種方式是把感測系統放在鞋墊裡，當受試者走動時，得以紀錄腳底資訊。

第一節 地毯型感測系統

[1]地毯型的腳壓感測器是一種將壓力感測器排列，放置於長方形地毯之中，系統能夠監測受試者行走時的壓力變化，從受試者的步伐中提取特徵，並將它變成辨識資料。從裡面的實驗中可以發現，感測器的佈置面積充滿整個地毯，在 180 公分乘以 80 公分的地毯上，長邊放了 120 個傳感器，短邊放了 54 個傳感器，也就是總共使用 6480 個傳感器。這樣做不僅成本很高，感測數據也會因為鞋子的種類和受試者行走速度的影響而失準，進而影響感測的精準度。

根據現況的研究，某些是傾向於使用大範圍的感測平台，[1]像是圖 1 將感測器鋪滿整個地毯(a)，然後再蓋上約 5mm 厚的普通地毯(b)，這個長型的感測系統會監測使用者的雙腳足跡變化，以及壓力變化，並從中取出特徵。但是在訓練時，因為感測器數量太多，而且收集到很多的足跡變化，所以會讓訓練的時間拉長許多。

第二節 鞋墊型感測系統

如圖 2 在鞋墊中建置感測系統，使用者在平時走動的時候，感測器就會收集壓力資訊。這種方式可以根據每個受試者的腳的大小，去客製化專屬鞋墊。此方式獲取到的內容則屬於比較長時間的資訊，因為感測器會長時間的放置在鞋裡，不過雖然這種感測系統有不受鞋種的影響的優點，但感測器長時間的放置在鞋裡，難免會受到擠壓、溫度以及濕度的干擾，很難保證感測器不受影響而讓辨識率降低。

將感測系統建置於鞋墊裡，或是鞋子中，其目的在於偵測使用者的日常行為動作與生活，能夠感應姿勢與壓力變化來判斷人的行為。[2]像是圖 2 便透過鞋墊內的感測器，來感測使用者的行為，並加以分類，包括走路、站著和坐著。對於這樣的分類，模型的辨識可以達到 95.4% 的精準度，這代表這套壓力感測設備系統已趨於能夠辨認人體動作，若以此進一步作為辨識人的依據上，有很強大的可行性。



(a)



(b)

圖 1 地毯式感測系統，取自[1]



圖 2 鞋墊式感測系統，取自[2]

第三節 現況探討

根據現況的研究，感測器在接收資料後透過特徵提取，並使用 SVM (Support Vector Machine) 進行訓練，根據[3]的壓力辨識所獲得的辨識結果還算不錯，測試者只需穿著附有感測器的鞋子正常走動，走 3 步就能達到 91.9% 的準確度，走 6 步更是能達到 95.2% 的準確度。

另一種像是論文[4]，先測得數據並轉換為圖像後，使用 CNN (Convolutional Neural Network) 進行學習，其好處在於前處理比較快，該論文還進一步結合 RNN (Recurrent Neural Network) 訓練，通常 CNN 與 RNN 的訓練時間更長，但是訓練出來的辨識率，準確度能夠達到 87.6%。

就目前的論文現況看來，辨識的準確率和學習時間還需要達到更好的平衡，而且很多論文也較少談到收集與訓練資料所需的成本，許多的論文也對於系統的訓練與辨識的時間研究較少。由於現代的辨識系統，像是臉部辨識、虹膜辨識、指紋辨識、悠遊卡辨識等，都能夠做到從感測到辨識只需要不到 1 秒的時間。所以辨識系統需要及時處理資料以及辨識，這時處理速度就會顯得重要。加上 CNN+RNN 在機器學習的訓練時間較長，訓練與辨識的時間成本也應該要加入考慮，這樣對於取得準確率和學習時間的更好之平衡也有幫助。

總結來說，就是目前的研究下，收集資料較不易，因為感測系統的成本比較昂貴。然後用來訓練的 SVM 與 CNN+RNN，雖然這些演算法的辨識表現不錯，但是它們需要的學習時間很長，所以我便想嘗試其他的機器學習演算法，並找出表現能夠比 SVM 與 CNN+RNN 來的更好，也就是學習速度更快且資料辨識的準確度更高的演算法。

第三章 研究方法及步驟

第一節 開發工具與軟硬體需求

在這個系統的整個流程裡，我們首先需要取得腳底壓力的相關數據，所以我們需要能夠感測壓力的感測器，以搭建一個如圖 3 的感測平台，在平台上面，將 40 個感測器排列成長為 10 個、寬為 4 個的長方形分佈。再來是 Arduino，Arduino 用於採集 40 個感測器感測到的電壓值，將這些電壓值的資料轉換成壓力值，並傳送到電腦上的資料庫儲存。電腦會對壓力值擷取出一些特徵並儲存起來，接者將整理出特徵的資料、數據使用機器學習進行訓練、建立模型。在模型建立完成之後，系統便會得到辨識的準確率以及模型的訓練時間。

所以在硬體的需求上，需要感測器、Arduino Nano、電腦，而軟體方面只需要能執行 Python 程式的編譯器即可，因為我是使用 Python 這個語言來進行機器學習運算的程式編寫。而我用於機器學習的開發工具選擇了 Apple 的 Macbook Air，搭載的是 M1 處理器，作業系統為 macOS Monterey 12.3。

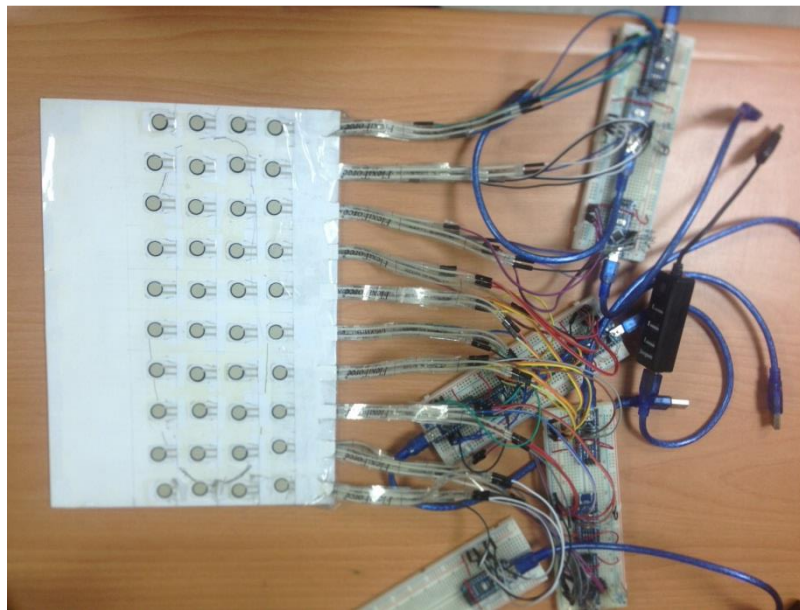


圖 3 感測平台

第二節 系統結構與操作流程

整個系統的流程為一開始受試者先以自然的步伐走過由感測器搭建的硬質平台，這 40 個感測器在這時會使用電壓值記錄壓力數據，經過 Arduino nano 將電壓值轉換成以重量表示的壓力值，並傳送資料給電腦的資料庫，這時電腦對資料庫擷取特徵，並使用機器學習進行辨識的訓練，記錄下訓練所花的時間並建立模型，最後再透過檢驗得到辨識準確率。

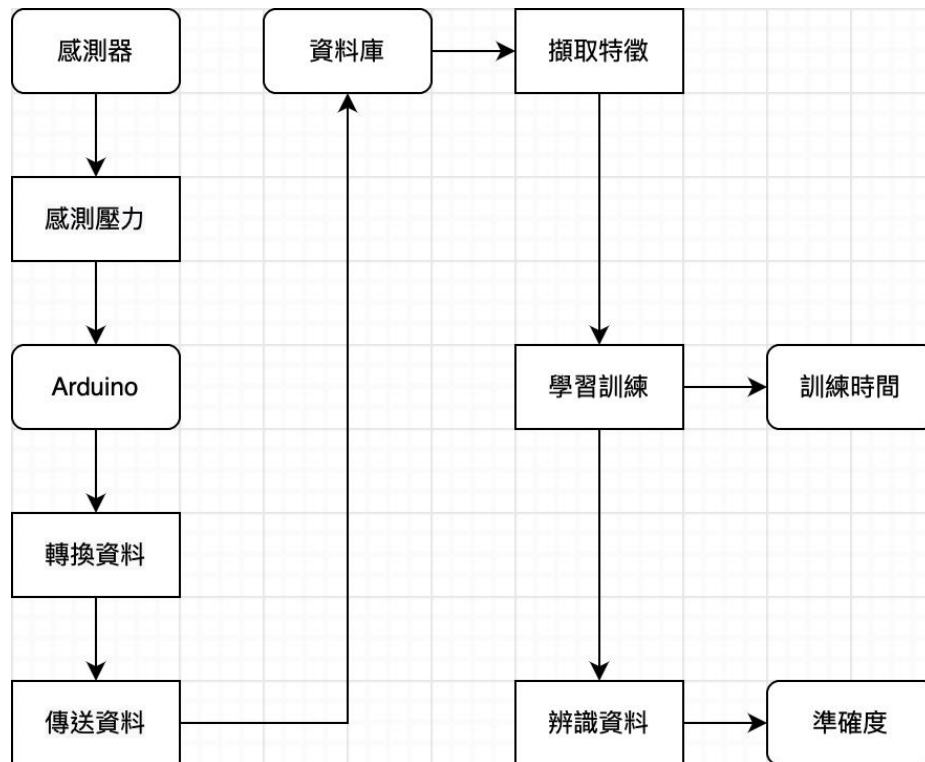


圖 4 系統流程圖

第三節 機器學習

在機器學習運算的部分，使用 5 個不同的分類演算法，分別為 SVM、RF、GB、kNN、XGBoost。將資料的 80% 作為訓練集、剩下的 20% 作為測試集。

其中 RF、GB、XGBoost 都是屬於集成學習(Ensemble learning)演算法的一種，集成學習指的是以一個系統化的方式將好幾個模型結合在一起。在許多科學競賽中 Ensemble learning 在實務上是非常有效的提升準確率。依照 Ensemble 的處理方式的不同，我們可以將它分為三類，Bagging、Boosting、Stacking。

3.3.1 SVM

SVM，也就是 Support Vector Machine（支援向量機）的簡稱，在機器學習中用於「分類」的演算法裡算是常見的，近幾年也被許多和機器學習有相關的研究與論文使用。論文[3]使用 SVM 來辨識受試者穿著有安裝感應器於鞋底的鞋子，受試者只需走 2 步，就能讓 SVM 的辨識達到 89% 的準確度，走 6 步更是達到 95.2%，分析的結果相當不錯，所以選擇 SVM 作為其中一種辨識方法。

3.3.2 RF

RF，也就是 Random Forest（隨機森林）的簡稱，為集成學習(Ensemble model)演算法的 Bagging 方法之一，會針對之前訓練錯誤的結果加強權重，除了取整筆壓力數據外，也可能

取不同的特徵，RF 在對於多特徵的資料也能達到很高的準確度。由於我們會對腳底的壓力資料擷取許多特徵，而且在訓練與辨識上的速度也相當快，所以也選擇 RF 作為其中一種辨識方法。

3.3.3 GB

GB，也就是 Gradient Boosting 的簡稱，為集成學習演算法的 Boosting 方法之一，其學習的方式是對前一次訓練錯誤的資料加強權重，在下一次的訓練中就可以優先改善錯誤的部分，藉此提升模型的準確度。論文[5]中使用 Gradient Boosting 辨識腳底壓力的資訊，在他的原始資料中，辨識的準確度高達 89%，且訓練的時間也短，所以也選擇 GB 作為其中一種辨識方法。

3.3.4 kNN

kNN，也就是 k-Nearest Neighbor 的簡稱，這個演算法會設定 k 值，當演算法想要知道這筆腳底壓力的數據是由哪位受試者所產生的，它會搜索和這筆數據最相近的 k 筆資料，並從這 k 筆資料去統計哪一位受試者的資料在這 k 筆資料裡最多，然後將最多的那位受試者當作是這筆資料的主人，這個演算法由於僅需要計算出每筆資料的最近 k 筆資料為哪些，所以學習的時間非常快。論文[3]使用 kNN 來辨識受試者穿著有安裝感應器於鞋底的鞋子，受試者只需走 3 步，就能讓 kNN 的辨識達到 88.6% 的準確度，走 5 步更是達到 93.3%，分析的結果相當不錯，所以也選擇 kNN 作為其中一種辨識方法。

3.3.5 XGBoost

XGBoost，也就是 eXtreme Gradient Boosting（極限梯度提升）的簡稱，為集成學習演算法的 Boosting 方法之一，在 Kaggle 網站上的機器學習相關競賽裡，辨識表現最頂尖的那些選手，大部分都是使用此演算法進行訓練，這個演算法除了擁有 Boosting 方法的高準確度的特性，而且其訓練速度也非常快，訓練時間只需要 Gradient Boosting 的數十分之一，所以也選擇 XGBoost 作為其中一種辨識方法。

第四章 效能評估

第一節 設計實驗與特徵設定

這一節會提到我們要怎麼運用機器學習進行實驗，包含演算法對資料的系統辨識率的關係、不同演算法的訓練時間差異，以及不同特徵對系統辨識率的關係。

我使用的壓力數據是來自論文[5]，他總共收集了 15 個人的腳底壓力，測得的原始資料共有 15000 筆(15 人*2 左右腳*25fps*20 次)，每個人使用左右腳各走過 20 次感測器。在特徵的部分，每筆資料的特徵分別為受試者編號、左右腳、某隻腳走的第幾次、40 個感測器測得的壓力值。

除了這些特徵之外，我們額外設定一些特徵，想要藉此觀察這些特徵對於原本資料的準確度是否有影響，以及影響多少。所以我們藉由觀察資料來額外提取長寬以及重心這兩個特徵。

4.1.1 長寬

我們認為受試者的腳掌大小也能當作是辨識的特徵，可以用感測器的感測狀況來推導出，也就是我們可以計算當受試者的腳正壓在感測器上面時，有哪些感測器有感應到重量。

我們將 40 個感測器擺放成如圖 5，長邊的計算方式是看短邊的每排 4 個感測器是否有至少 1 個感測器的壓力值不為 0。如果有的話長邊+1。短邊的計算方式相同，看短邊的每排 10 個感測器是否有至少 1 個感測器的壓力值不為 0。如果有的話短邊+1。

	1	2	3	4
	5	6	7	8
	9	10	11	12
	13	14	15	16
	17	18	19	20
	21	22	23	24
	25	26	27	28
	29	30	31	32
	33	34	35	36
	37	38	39	40

圖 5 40 個感測器之分佈

舉某 2 筆資料為例，第一筆資料在 4 個長邊裡，第 1 個長邊有 6 個感測器的壓力值不為 0，第 2 個長邊有 7 個感測器的壓力值不為 0，第 3 個長邊 7 個感測器的壓力值不為 0，第 4 個長邊的感測器壓力值全為 0，所以短邊為 3。同理，10 個短邊裡有 8 個邊是感測器的壓力值不為 0 的情況，得到長邊為 8。而第二筆資料的所有短邊和長邊均有感測器感應到重量，所以短邊為 4、長邊為 10。

```
length: [6, 7, 7, 0]
width: [0, 0, 3, 3, 3, 2, 2, 3, 2, 2]
len-on-excel: 8.0
width-on-excel: 3.0
length: [3, 8, 8, 3]
width: [3, 4, 4, 3, 1, 1, 2, 2, 1, 1]
len-on-excel: 10.0
width-on-excel: 4.0
```

圖 6 提取長邊及短邊特徵的過程

4.1.2 重心

由於每位受試者的腳底重心位置不太一樣，所以也能當作是辨識的特徵，將上面提到的短邊當作 X 座標，長邊當作 Y 座標。我們定義 $G_x(A(ID, LERI, Order, X, Y))$ 代表在第 ID 位受試者時，左腳或右腳（LERI 分別是 LE 為左腳，用 1 表示。RI 為右腳，用 2 表示。）的第 Order 次，座標在 X, Y 的平均壓力數值的 X 軸重心座標，而 $G_y(A(ID, LERI, Order, X, Y))$ 則代表在第 ID 位受試者時，左腳或右腳（LERI 分別是 LE 為左腳，用 1 表示。RI 為右腳，用 2 表示。）的第 Order 次，座標在 X, Y 的平均壓力數值的 Y 軸重心座標。

圖 7 為重心的計算公式，令 $B=A(ID, LERI, Order, X, Y)$ 。

$$G_x(B) = \frac{\sum_{x=1}^4 x (\sum_{y=1}^{10} B)}{\sum_{y=1}^{10} \sum_{x=1}^4 B}$$

$$G_y(B) = \frac{\sum_{y=1}^{10} y (\sum_{x=1}^4 B)}{\sum_{y=1}^{10} \sum_{x=1}^4 B}$$

圖 7 重心的計算公式

4.1.3 訓練樣本設定與準確度判定標準

在原始資料中，一個步伐會產生 25 筆資料，平均資料則是將原始資料裡每次走過的 25 筆資料取平均成一筆，也就是變成 600 筆。然後將這 600 筆平均後的資料分成 80% 用來訓練模型，並以剩下的 20% 用來預測、辨識，這邊我是將每位受試者的左右腳的前 16 次記錄用來訓練，剩下的 4 次用來辨識，並以混淆矩陣(confusion matrix)的結果當作是系統辨識的準確度。

因為這份資料是由 15 位受試者產生的，我們的資料就會分成 15 個分類，所以混淆矩陣會長的像是圖 8，為 15x15 的二維矩陣，其中藍色區域表示機器辨識的結果和實際結果是同一个人，代表這筆資料被正確地辨識。而橘色則是機器辨識的結果和實際結果為不同人。我們的系統辨識率算法如圖 9。

		預測				
		受試者1	受試者2	受試者3	...	受試者15
實際	受試者1					
	受試者2					
	受試者3					
	...					
	受試者15					

圖 8 混淆矩陣示意圖

$$\text{系統辨識率} = \frac{\text{藍色區域總和}}{\text{藍色區域總和} + \text{橘色區域總和}}$$

圖 9 系統辨識率算法

第二節 演算法對資料的辨識準確度與訓練時間

4.2.1 平均資料

圖 10 為使用 5 個演算法(SVM, RF, GB, kNN, XGBoost)對於平均資料進行資料預測的準確度，將這 15 位受試者的每隻腳所踩的前 16 步用於學習，第 17~20 步作為資料預測，讓模型對這些預測資料預測完後，再與實際的第 17~20 步資料進行混淆矩陣的比對，並算出準確度。

以下數據為每個演算法各運算 3 次，然後對 3 次的準確度取平均。我們可以發現屬於集成學習演算法的 RF, GB, XGBoost，資料預測的準確度都非常高，在這 3 次裡，GB 都是達到 98.3% 的準確度，而 XGBoost 都是 100%，RF 則是落在 95.8~97.5% 之間。

至於 SVM 和 kNN 準確度不太理想的原因，應該是和演算法本身的特性有關。SVM 會根

據特徵數建構一個多維度的空間，平均資料共有 40 個特徵，也就是 40 維的空間。SVM 建構分類的標準為在兩個分類中用一個超平面最大化的區隔，而這份資料共有 15 個分類（15 位受試者），因為這份資料的維度和分類數對於 SVM 來說太複雜，所以在這 3 次的辨識都是只有 60% 的準確度。

而 kNN 則是某筆資料由最相近的 k 筆資料，來決定這筆資料的分類。但是這份平均資料的壓力數據，和受試者的體重、這一步在感應器停留的時間有高度相關，所以很有可能產生好幾筆資料非常相近，但是卻來自完全不同的受試者，使得分類的決定變得混亂、棘手，所以在這 3 次的辨識都是只有 50.8% 的準確度。

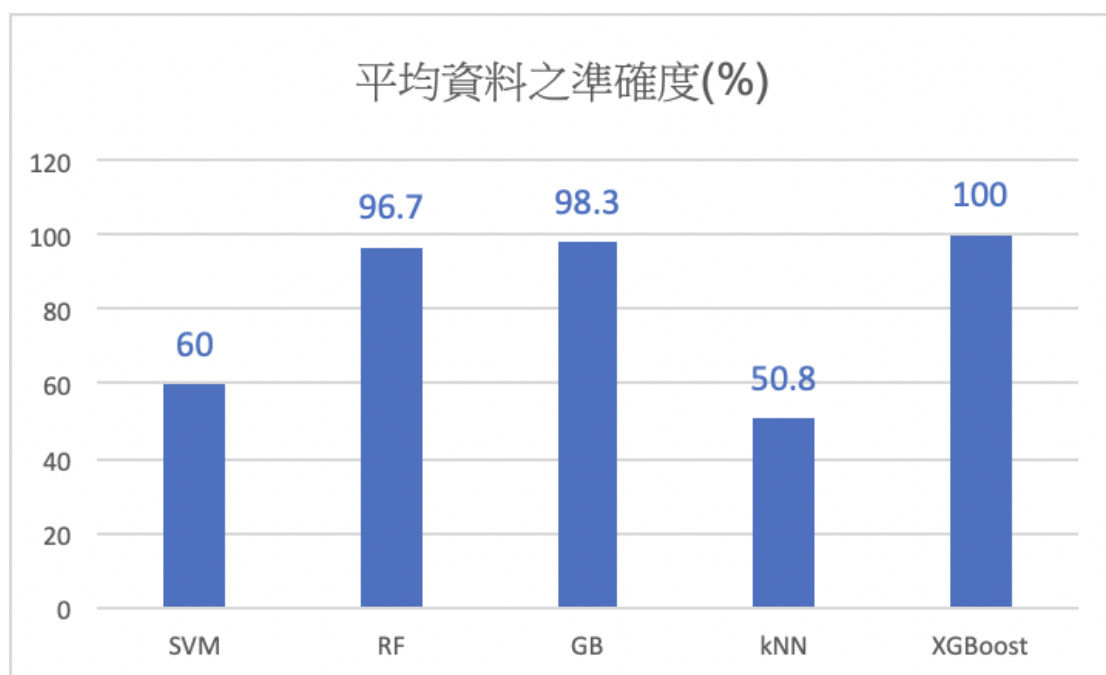


圖 10 不同演算法預測平均資料之準確度(單位：%)

接者我們來看圖 11，也就是 5 個演算法(SVM, RF, GB, kNN, XGBoost)對於平均資料進行學習所花的時間。我們可以發現，如同第三章第三節介紹各個演算法的時候所提到的，XGBoost 的訓練時間約為 GB 的 12.62 分之 1，確實是一個訓練快、辨識準確度也很高的演算法。而 RF 只比幾乎不需要訓練的 kNN 慢，所以也是一個訓練快、辨識準確度高的演算法。

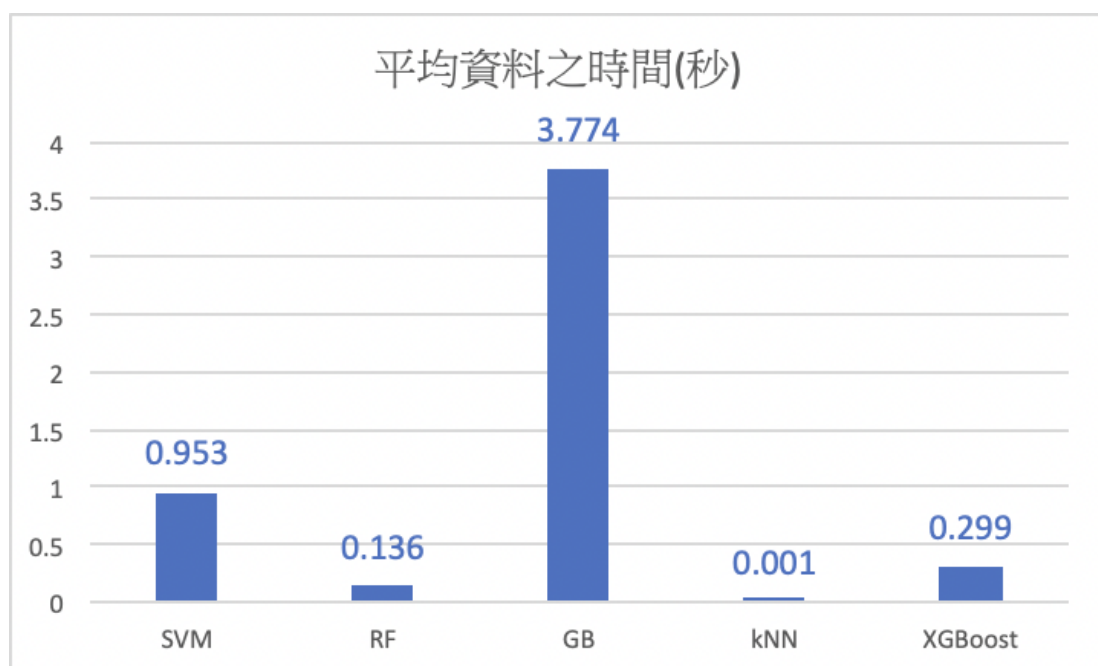


圖 11 不同演算法對平均資料學習之所需時間(單位：秒)

4.2.2 原始資料

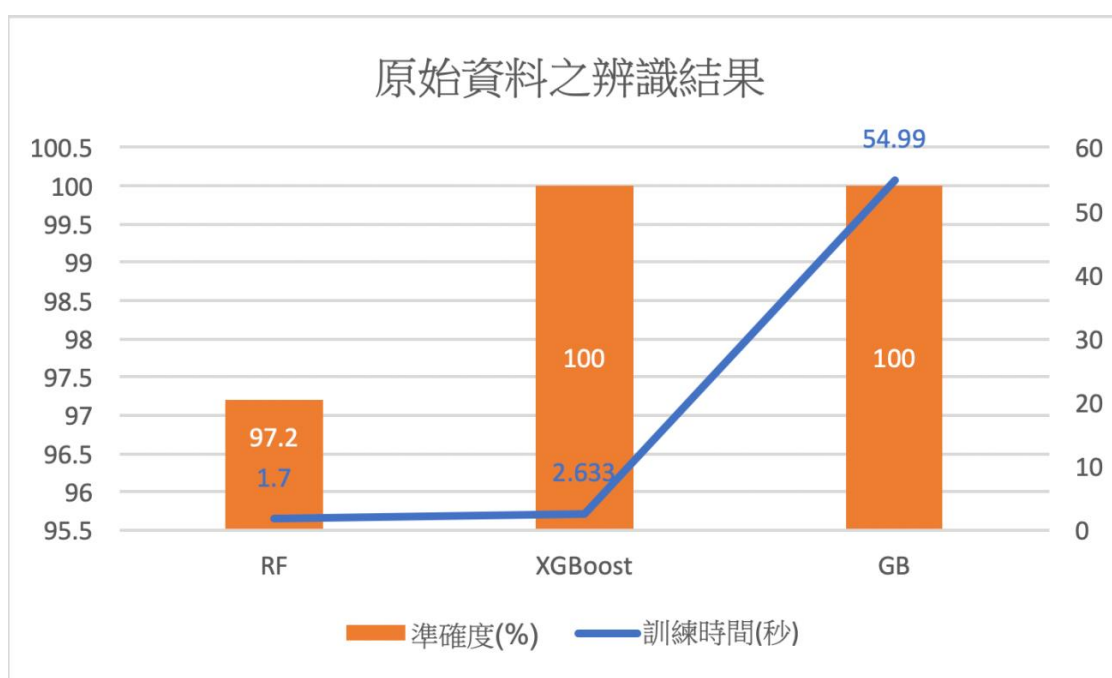


圖 12 不同演算法預測原始資料之準確度與訓練時間

接著使用 RF, GB, XGBoost 這 3 個在平均資料上表現特別好的演算法，改用原始資料進行模型訓練和資料辨識。一樣將這 15 位受試者的每隻腳所踩的前 16 步用於學習，第 17~20 步作為資料預測，讓模型對這些預測資料預測完後，再與實際的第 17~20 步資料進行混淆矩陣的比對，並算出準確度。

結果如圖 12，即使測試資料從 $600 \times 0.2 = 120$ 筆（1 步 1 筆）變成 3000 筆（1 步 25 筆），這 3 個演算法的預測表現依然非常頂尖，而且 GB 在原始資料的表現甚至比平均資料還要好，

能夠完全正確的預測受試者的身份。

第三節 特徵與辨識準確度的關係

圖 13 為 5 個演算法(SVM, RF, GB, kNN, XGBoost)對於含有不同特徵的資料進行資料預測的準確度，可以發現在之前表現沒這麼好的 SVM, GB 在新增重心、長寬等特徵後，資料預測的準確度有明顯提升，而另外 3 個集成學習演算法則是保持優異的預測表現。

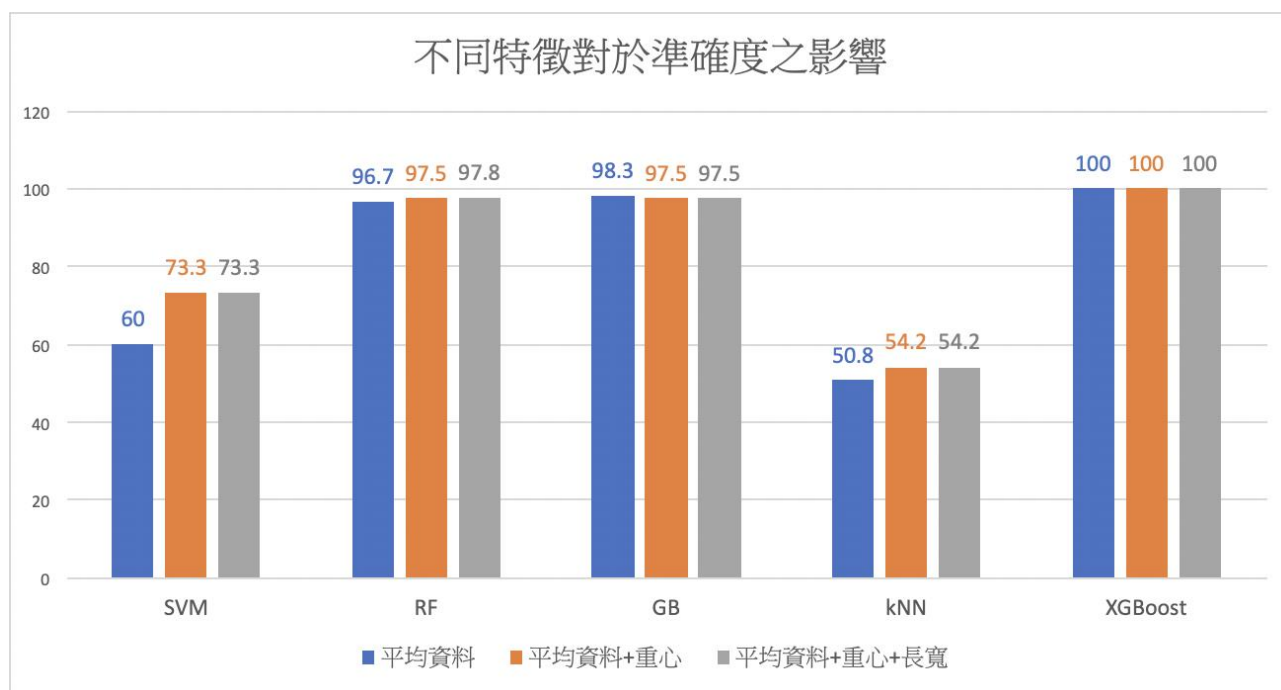


圖 13 不同特徵對於準確度之影響

第五章 結論

第一節 各組員的貢獻

這份專題是由我一個人負責的，實驗、書面報告、簡報這些全都是我的貢獻，但我要感謝我的指導教授，他會適時地提供研究方向，讓我自己去選擇我的下一步。

第二節 後續研究方向

本文想要建構一個辨識準確度高、收集便利、成本低且訓練速度快的腳底壓力辨識系統，其中使用 RF 與 XGBoost 這兩個演算法的效果最好，GB 的準確度也很高，但是在訓練的速度上沒有 RF 與 XGBoost 來的出色，所以 RF 與 XGBoost 會是這個辨識系統的機器學習演算法的首選。

這些資料是基於受試者沒有穿鞋子的情況下測試出來的，而且身上沒有額外的負重。所以希望在未來能夠考量更多會影響數據的條件，比如讓受試者穿上鞋子測試，或是讓受試者背個幾公斤的重物測試。

參考文獻

- [1] Bo Zhou, Monit Shah Singh, Sugandha Doda, Muhammet Yildirim, Jingyuan Cheng and Paul Lukowicz, "[The Carpet Knows: Identifying People in a Smart Environment from a Single Step](#)," *IEEE International Conference on Pervasive Computing and Communications Workshops*, Kona, HI, USA, 13-17 Mar. 2017.
- [2] Nagaraj Hegde, Edward Melanson and Edward Sazonov, "[Development of A Real Time Activity Monitoring Android Application Utilizing SmartStep](#)," *IEEE Annual International Conference on Engineering in Medicine and Biology Society (EMBC)*, 16-20 Aug. 2016, pp. 1886-1889.
- [3] Gu-Min Jeong, Phuc Huu Truong, and Sang-Il Choi, "[Classification of Three Types of Walking Activities Regarding Stairs Using Plantar Pressure Sensors](#)," *IEEE Sensors Journal*, Vol. 17, NO. 9, May 2017, pp. 2638-2639.
- [4] Monit Shah Singh¹, Vinaychandran Pondenkandath, Bo Zhou, Paul Lukowicz and Marcus Liwicki, "[Transforming Sensor Data to the Image Domain for Deep Learning - an Application to Footstep Detection](#)," *IEEE International Conference on Neural Networks*, Anchorage, AK, USA, 14-19 May 2017, pp. 2665-2672.
- [5] 陳建翰, "腳底壓力辨識系統結合機器學習之分析與研究," 臺灣師範大學資訊工程學系學位論文, pp. 1-30, 2019.
- [6] 楊瑀婕, "腳底壓力辨識系統對於受測者在不同負重支撐點與重量之分析與研究" 臺灣師範大學資訊工程學系碩士論文, pp. 1-39, 2019.