東海大學

資訊工程研究所

碩士論文

指導教授: 林祝興博士

利用類神經網路分析以擊鍵動能為基礎的生物特徵驗證

On the Neural Networks for Biometric Authentication Based on Keystroke Dynamics

研究生: 李耕瑜

中華民國 106 年 7 月

東海大學碩士學位論文考試審定書

東海大學資訊工程學系 研究所

研究生 李 耕 瑜 所提之論文

利用類神經網路分析以擊鍵動能為基礎的生物特徵驗證

經本委員會審查，符合碩士學位論文標準。

學位考試委員會

召 集 人 蔡英德 簽章

委 員 劉榮春

江憶恩

蔡英德

指 導 教 授 [signature] 簽章

中華民國 106 年 7 月 14 日

摘　要

隨著科技快速發展，密碼跟人們的關係越來越密切，但科技的發展同時也增加了密碼外洩的風險，GPU 平行化暴力破解或生日攻擊演算法更下降了密碼的安全性。此外使用者也常常透過通訊軟體傳送密碼，通訊軟體時常會忘記登出或遭受攻擊，使得有心人士取得密碼的機會大幅提升。因此本論文將提出一種身分驗證法，在即使知道完整密碼的情況下，依然可以阻擋大部分的非法使用者。本論文在使用者輸入密碼時，同時側錄了使用者的鍵盤碼和按鍵順序，接著利用類神經網路的機器學習，可有效的區分出合法使用者和非法使用者。實驗結果顯示，即使使密碼在已經外洩的情況下，依然可以阻擋 100%的非法使用者，雖然合法使用者遭到誤判的機會大約是 6%，但若使用者是合法的，只需再輸入一次密碼即可。實驗中我們比較了卷積類神經網路 (CNN)、類神經網路 (NN) 的辨識率，證明了採用卷積神經網路是優秀的選擇，並且透過調整參數以提高準確率。 最後為了日後實際應用上的需要，本論文加入了 GPU 平行化運算，並且達到了近 5 的倍加速比。

關鍵字: 生物特徵、擊鍵動能、機器學習、深度學習、類神經網路、卷積神經網路、GPU 平行化

Abstract

In this study, we propose a biometric authentication method to identify and block illegal users, even if the whole password is exposed. Our method simultaneously records scan codes and keystroke sequence of passwords; furthermore, by deep learning of convolutional neural networks, legal users can be effectively distinguished from illegal users. The experimental results show that illegal users are successfully blocked even if the password has been exposed. Although the average login failure rate of legal users is 6 percent, they can reenter passwords once to be admitted. We also compare recognition rates between convolutional neural networks and neural networks and prove that convolu- tional neural networks are better. Finally, by GPU parallel computing, we further obtain about 5 times acceleration of system performance.

Keywords: biometric authentication, keystroke dynamics, machine learning, deep learning, convolutional neural network, GPU parallel computing

致謝

兩年研究所的日子轉眼間就要結束了，這些日子以來有許許多的人們需要感謝，感謝有你們的陪伴，幫忙與諒解。

首先需要感謝的是在大學時就幫助我許多，從大學專題到研究所的指導老師 林祝興老師，願意讓我在研究所時嘗試這麼特殊的議題，真的非常的感謝老師一路上的包容與幫助。

李耕瑜 於東海大學研究所 資安實驗室　106 年 7 月

總目錄

圖目錄

表目錄

公式目錄

第一章　簡介

在這個科技發展快速的時代，密碼早就和生活形影不離，需要密碼的服務充斥在人群周遭，如 Figure 1.1 中的許多密碼服務對人們更是至關重要，一旦密碼外洩很有可能損失大量財務或資訊，因此擁有安全的密碼逐漸受到人們的重視。但科技的發展的 同時也加速了密碼的破解，例如:GPU 的平行化運算暴力破解法和生日攻擊演算法，這兩種演算法具有極高的效能，相較於傳統的演算法提升了成百上千倍的破解速度，使密碼的安全性大幅下降。為了對抗這些演算法的出現，目前普遍的方法是增加密碼的複雜度和長度，可惜的是任何複雜的密碼都有可能因為使用者的疏忽而外洩。因此本論文提出一種身分驗證法，即使在知道完整密碼的情況下，依然可以阻擋大部分的非法使用者，使密碼更有保障。

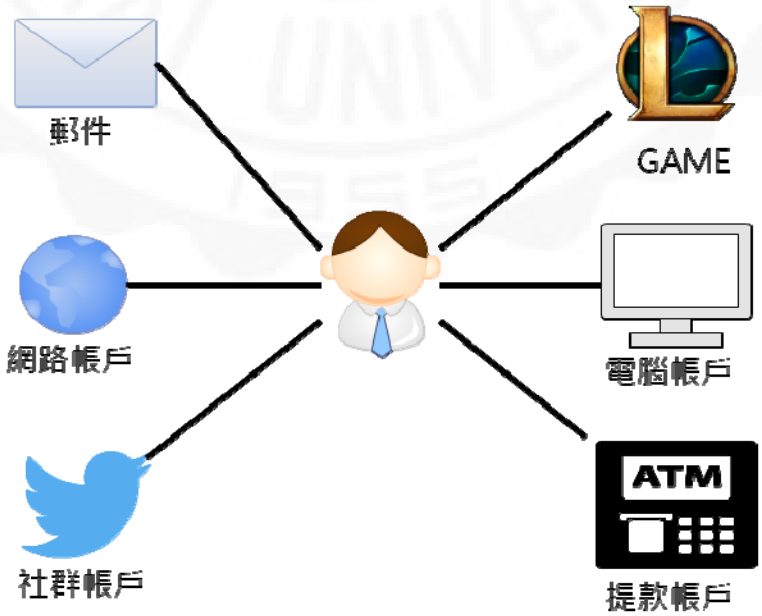

Figure 1.1: 說明密碼的使用普遍應用於生活當中

為了增強密碼的安全性，並且在不增加密碼的複雜度的情況下，本論文在使用者輸入密碼時多加入了一個側錄程式，透過這個程式可記錄使用者的打字習慣，再利用每個人打字習慣的差異當作生物特徵 [2]，以機器學習來區分合法使用者和非法使用者。其中打字習慣乍看之下差異極小，但實際上並非如此，由於使用者長時間輸入某幾組密碼，往往會對自己的密碼擁有一套熟練的打字習慣 [2],[3]，這種習慣有可能是某兩個鍵的下壓時間差，亦或是習慣 shift 和右 shift 的按鍵差異，這些都使得密碼更具獨特性，利用這具有獨特性的習慣，即可區分出合法使用者和非法使用者。然而這些生物特徵是很難被肉眼看出來的，在程式設計中更難以用一般判斷法來區分，因此本論文借助機器學習的強大特徵尋找能力，透過多次學習訓練，最終能有效的區分出合法使用者和非法使用者。[4-5]

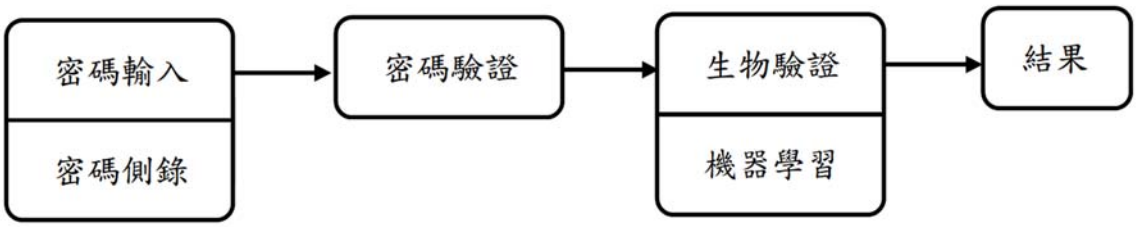

Figure 1.2: 說明本論文所提方法的示意圖

機器學習是一種分析資料且預測結果的演算法模型，有著強大的決策分析能力，有別於傳統的程式判斷演算法，機器學習能從學習樣本中找出資料間的規律，進而找出資料的特徵，再利用此特徵分類或歸納，使機器能依照不同的樣本作出不同的輸出，被廣泛使用在各種領域，例如：影像辨識、人工智慧、數據預測和自然語言分析等等。本論文探討了兩種不同的機器學習模型，類神經網路 (neural network, NN) 和卷積神經網路 (convolutional neural network , CNN)，這兩種模型都可利用輸入的特徵來分析或預測結果，差別是卷積神經網路在分析特徵時，會將多個特徵串聯起來，使特徵彼此間具有關聯性，如此可大幅提升分析或預測的準確度。但卷積神經網路經常因為學習力過強導而致過度擬合 (overfitting)[10,12]，過度擬合是指機器學習中，機器過度的學習，導致分析函數曲線扭曲，使最終結果不符預期。為了避免這個情況，本論文中加入了 Dropout regularization[13]，可適度的忽略掉部分的神經元權重，降低發生過度擬合的機會，使輸出更加準確。類神經網路的機器學習有著優秀的分析決

策能力，完美的解決了傳統條件判斷式無 法分析的問題，可惜的是機器學習經常因為大量次數的學習而浪費許多時間，為了降低學習時間，本論文加入了 GPU 平行化運算。

GPU 是圖形處理器(graphics processing unit)，過去經常用來處理影像的運算，但隨著近幾年來資料 量的龐大和運算的複雜，人們發現 GPU 在處理這些資料時有著優異的表現，其原因在於 GPU 有著大量的運算核心，每個核心內又具有大量的執行續，並且運算時每個核心能獨立進行。有別於 CPU 只能進行分時多工的運算，GPU 可以大幅提升運算速度，尤其在矩陣的運算效能更是明顯，機器學習具有大量神經元和權重，這些都是存放在陣列內，而陣列剛好適合矩陣的形式，因此 GPU 平行運算可以完美幫助機器學習，降低運算時間，所以本論文在訓練資料時加入 GPU 平行化運算技術，改善運算效能。

在實驗中，本論文為求數據的真實性，採樣了十名不同的受測者，從而進行分析，不僅比較了類神經網路 (NN) 和卷積神經網路 (CNN) 的分析預測能力，更利用所採樣的數據一一分辨出十名不同的使用者，證明了生物習慣具有獨特性。實驗的結果在 25000 次學習後，準確度可以達到 99%，其中非法使用者的阻擋率高達 100%，使密碼的安全性大幅提升。本論文也測試了一個比較符合現實的情境，在實際使用上，非法使用者的學習資料通常是難以取得的，因此一個好的身分驗證法必須要在無學習資料的情況下分辨出合法使用者，所以論文也做了一個實驗，僅利用一個非法使用者和一個合法使用者的樣本當作學習資料，在 18 人樣本中找出正確的合法使用者，實驗的結果也達到 80% 以上，代表本方法至少能增加 8 成密碼的安全性。最後本論文考量到日後應用，加入 GPU 平行化 運算，大幅優化效能 [11]，在 25000 次訓練學習時，可達到 5 倍的加速，使所提出的方法更加實用。

本論文共分為五個章節，首先透過第一章的簡介描述整個論文的內容，接著在第二章背景知識與相關文獻中詳細說明了使用到的技術，第三章中介紹了本論文所提出的方法設計，也介紹了所提方法的運作流程，第四章利用各種實驗證明了本方法的成效，最後在第五章中討論了本方法的優劣和未來的展望。

第二章　背景知識與相關文獻

2.1 機器學習

機器學習屬於人工智慧的範疇，是一種讓機器從「推理」到「學習」的過程，機器學習的演算法結合了各種領域，如：機率學、統計學、逼近論和計算複雜性理論等等，透過這些數學模型的運算可讓機器自我修正演算法推理的錯誤，並且從錯誤經驗中學習，找出更趨近於正確解答的演算法，在多次學習後的輸出結果具有高精準度的分析和決策能力，使機器學習在近幾年被廣泛使用在各種領域，常見的機器學習模型有類神經網路和卷積神經網路等等。

機器學習的理論早在二、三十年前就已經被科學家提出，但由於當時技術不佳，導致準確度不夠理想，因此很少人使用，近年來由於硬體效能的進步，並且許多演算法的提出，大幅改善了機器學習的預測準度，被廣泛利用在各種領域，舉凡影像辨識，自然語言分析，人工智慧，都能看到機器學習的身影，常見的機器學習演算法有類神經網路、基因法等等。

2.2 Neural Network

類神經網路是一種機器學習的演算法模型，設計者將程式的運算模擬成生物的神經網路傳遞，生物在傳遞訊號時，會通過數個神經細胞，最後傳達到大腦，而類神經網路的概念也類似如此，在類神經網路模型中具有大量神經元，神經元上具有一些參數，透過不斷的修改這些參數可讓最後傳遞出來的資料趨近一個預測值，進而達到分析和預測的效果。

類神經網路是由一組或多組類神經單元所組成，共可分為輸入層、隱藏層以及輸出層，輸入層負責接收或取得輸入的數值，這些數值會和神經元內的參數進行運算，參數通常包含權重(weight)和偏量(bias)，權重用來表示該神經元的價值，在類神經網路運算的過程中，會不斷修正權重，使整個模型輸出值趨近於解答，而權重越高則表示該神經元越能影響整個模型的結果，換言之表示該神經取得的特徵越重要。隱藏層是整個類神經網路的核心所在，也是主要影響準確率的地方，隱藏層可以是一個或數個，它的數量與解決問題的複雜度有關，一般來說隱藏層越多會使準確度越準，但運算的複雜度也會隨著提升。本論文中採用兩層隱藏層。輸出層是整個模型運算的結果，通常輸出的形式為線性回歸和分類，由於本模型是反向傳遞型類神經網路，輸出後的值會和正確解答運算後，逐步由後往前修正每個神經元的權重，進而提升整體準確度；線性回歸經常被用於趨勢分析，常見的分類使用在影像辨識。類神經網路有了這些架構，使它可以分析傳統判斷式無法分析的資料，並且有著優異的分析預測能力，詳細運作流 程如 Figure 2.1：

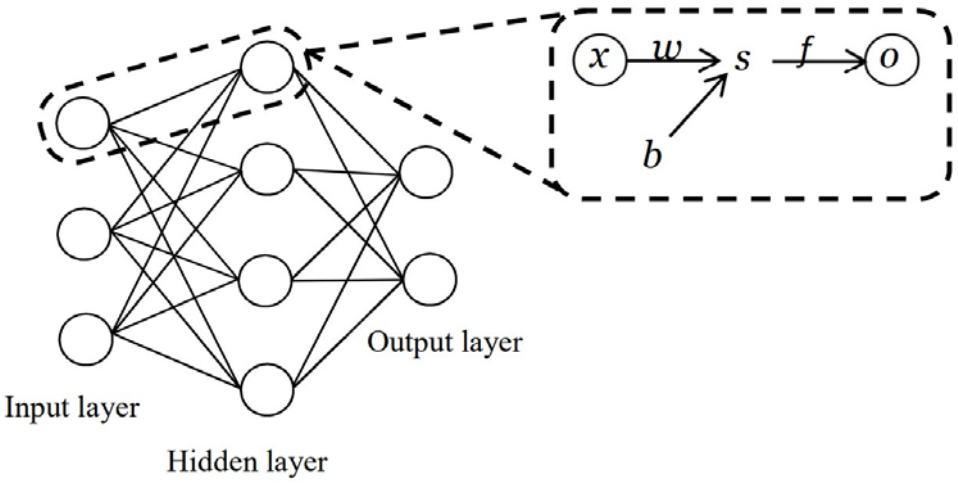

Figure 2.1: 類神經網路示意圖

Figure 2.1 中左邊為整個類神經網路的示意圖，圓形代表神經元，神經元兩兩之間透過數學運算互相連接，運算過程如同虛線區域表示，其中 x 為輸入向量的各個分量，共有 n 個輸入。w 為權重值，類神經網路最主要就是經由不斷的訓練、學習進而調整權重值。b 為偏量(bias)。s 為加法單元，每一個類神經元的輸入以及權重值相乘之後加總。f 為激勵函數 (activation function)，利用線性或是非線性的函數將 s 轉換成所需答案。o 為神經元輸出。

激勵函數是類神經網路中的一大重點，在早期的類神經網路很難解決特徵不明顯的問題，例如試著讓機器預測 XOR 的結果，若不使用激勵函數是幾乎不

可能有正確解答的，因為 XOR 在二維平面上並非一完整曲線 (直線)，但類神經網路卻只能預測線性結果，而激勵函數的價值就是能使類神經網路跳脫線性預測，讓整個預測結果更趨近真實，在本論文中所採用兩個激勵函數，其定義如Eq. (1) 和 Eq. (2)：

$$f(x) = \max(0, w^T x + b) \tag{1}$$

$$\sigma(\mathrm{w})_j = \frac{e^{w_j}}{\sum_{k=1}^{K} e^{w_k}} \quad for\ j = 1, \ldots, K \tag{2}$$

Eq. (1) 是激勵函數 ReLU，可改變神經元的閥值，當神經元的運算結果小於 0 時，本函數將其轉換為 0;大於 0 時則保持原本結果，由於值為 0 的神經元將不會對整個神經網路輸出造成任何影響，因此本函數可控制每個神經元在神經網路模型中的活性[14]。

Eq. (2) 是激勵函數 softmax，本函數透過自然對數轉換，使神經元的值轉換成介於 0~1 之間，如此一來便能比較出每個神經元的價值，對於分類或決策分析上有著不可或缺的重要地位，因此本論文選用此激勵函數來區分合法使用者和非法使用者[14]。

2.3 Convolutional Neural Network

是一種類神經網路為基礎的模型，對於影像處理擁有出色表現，其架構與類神經網路大致相同，不同的是多了一個或多個卷積層 (convolutional layer) 和池化層 (pooling layer)。

2.3.1 卷積層

每個卷積層由數個卷積單元組成，卷積單元會提取輸入矩陣的局部，提取的局部矩陣進入神經網路作分析，提取過程中可以加入權重來改變原始資料，如 Figure 2.2，圖中輸入矩陣為 3*3 的二微陣列，表格中的數字表示陣列內的值，卷積單元的範圍是 2*2，表格中的數字代表卷積單元的權重，卷積層在運作時會先提取輸入陣列的左上區域(藍色範圍)，提取的值會和權重相乘，接著相加產生出一個結果，$c_{11}=a_{11}\cdot b_{11}+a_{12}\cdot b_{12}+a_{21}\cdot b_{21}+a_{22}\cdot b_{22}$，依照這個運算式取得輸出，其中 c 是輸出矩陣，a 是輸入矩陣，b 是卷積單元，運算後產生圖中的 step 1，然後提取右上區域(紅色區域)，如同 step 1 的運作重複執行，直到整個輸入陣列完全提取，提取結束後，卷積層會將輸出陣列的厚度增加，厚度表示輸出陣列的多樣性，原始厚度為 1，增加厚度時，卷積層會亂數打散輸出陣列的結果，並且把所有打亂的輸出都合併為一個陣列，如此便可增加特徵的多樣性，輸出陣列的多樣性越高，越能讓類神經網路找出最能影響結果的特徵，也就是說多樣性能影響準確度。

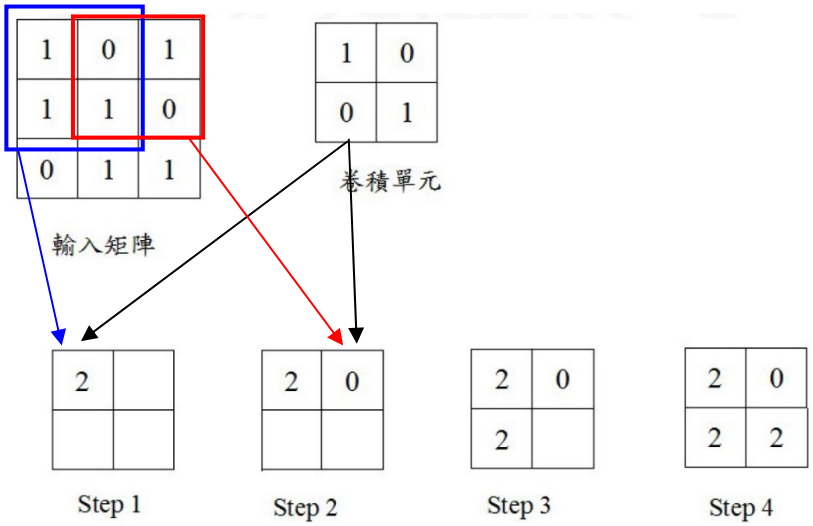

Figure 2.2: 卷積單元運作示意圖

2.3.2 池化層

卷積層中卷積單元一次移動的距離稱為跨步，跨步大可使輸入矩陣快速縮小，讓機器學習時的運算大量減少，但是過大的跨步會讓輸入的特徵訊息流失，為了避免大量流失特徵，卷積神經網路中加入了池化層，有了池化層之後，卷積層可以執行較小的跨步，並且透過池化層再次整理與縮小範圍，如此在確保了特徵完整性的情況下，大幅減少了運算量，Figure 2.3 表示了只有卷積層和加入池化層的差異。

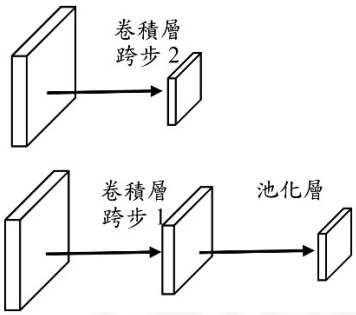

Figure 2.3: 卷積層和池化層示意圖

卷積神經網路中，卷積層負責凸顯特徵並減少運算量，池化層負責保留特徵避免特徵流失，在這兩層中又增加了特徵的複雜度，提升了整體特徵的多樣性，因此卷積神經網路加入了卷積層和池化層後，使整個網路在分析資料時不再是一個神經元一個特徵，而是一個神經元多個特徵，並且利用權重的改變，可以找出特徵之間的關聯性，從而大幅提升分析或預測的準確度。卷積神經網路被廣泛地運用在影像處理和自然語言分析等領域 [10],[12-13]。

2.4 CUDA

CUDA 是 NVIDIA 的平行運算架構，可運用繪圖處理單元 (GPU) 的強大處理能力，大幅增加運算效能。GPU 由多個多串流處理器 (stream multiprocessors , SM) 組成，每個多串流處理器內含有大量串流處理器 (stream processors , SP)，每個串流處理器都可獨立處理運算，因此 可以達到平行化處理資料，善於處理矩陣運算，特別是影像處理或視訊處理，可大幅加快計算速度，對效能提升有明顯的幫助。類神經網路通常具有大量神經元，每個神經元都獨立運算，所以可把神經元想像成矩陣圖像進行平行運算，改善效能 [15,18]。

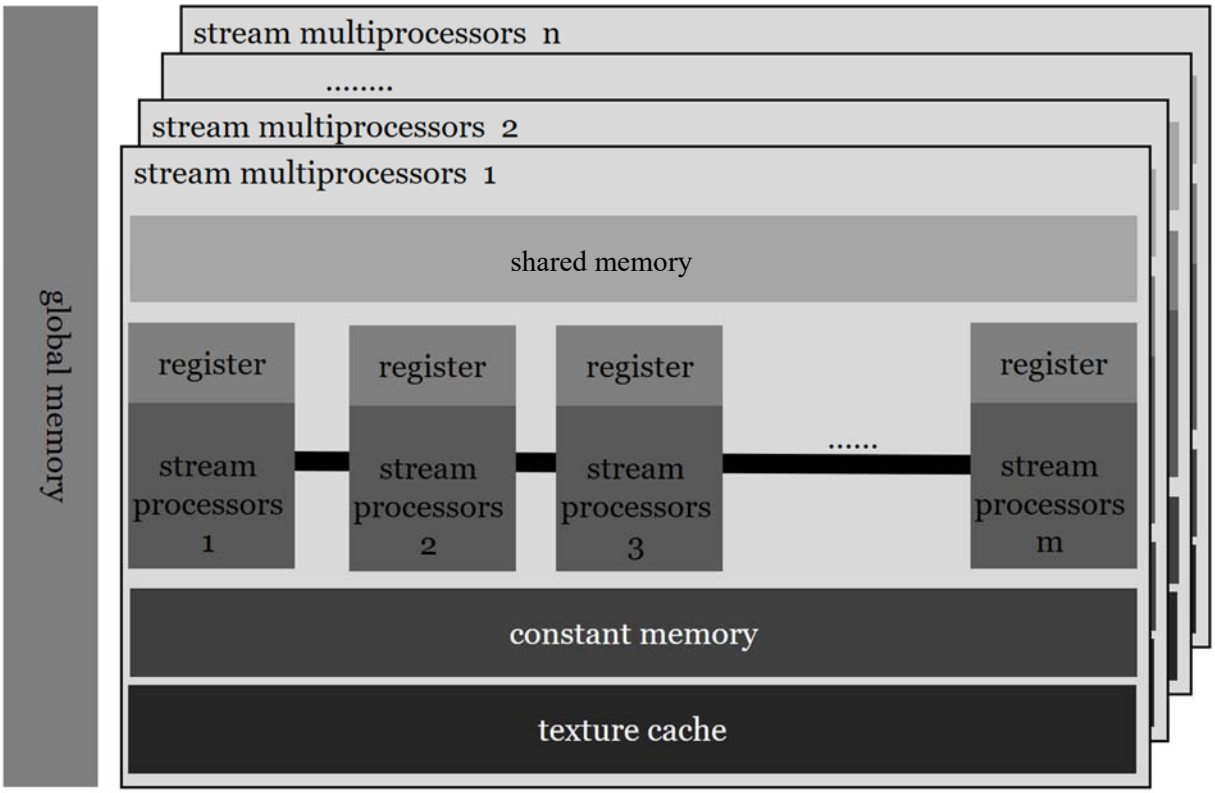

Figure 2.4: GPU 架構示意圖

Figure 2.4 表示 GPU 的架構簡圖，一個基本的 GPU 會具有大量的 SM 和一個全域記憶體(global memory)，每個 SM 中都具有大量的 SP，GPU 在執行時是由 SP 運算，SP 在讀取記憶體數值時，相同 SM 中的 SP 可藉由共享記憶體交換資料，但不同 SM 中的 SP 僅能透過全域記憶體交換資料，暫存器可提供 SP 來存放要使用的變數，由於暫存器容量不大，使用上經常超出負荷，超出容量的變數就儲存在常數記憶體(constant memory)，常數記憶體中的變數是唯讀的，在程式部屬時宣告完成就無法再修改，具有和共享記憶體一樣的存取範圍，材質記憶體(texture catch)通常做為繪圖時使用，也是唯讀記憶體，擁有快速轉換資料為浮點數的能力，由於每種記憶體的傳輸速度有著極大的差異，因此如何分配各種數值在不同的記憶體上是很重要的。

類神經網路中有大量神經元，神經元都有屬於自己的權重、偏量和值，因此在設計程式時，每個串流處理器當作一個神經元進行運算，權重、偏量和值放在 SM 內的共享處理器內，激勵函數 (activation function) 放在常數記憶體內，常數記憶體有著可以共享全部 SM 資料的特性，並且比全域記憶體的傳輸速度更快，缺點是必須在城市一開始就宣告記憶體數值，並且無法更改，因此拿來存放激勵函數相當合適，透過這樣的平行化模型設計， 可有效改善類神經網路效能。

第三章　基於類神經網路的生物特徵驗證

本論文改變了傳統密碼驗證的方式，不僅只比對密碼的完整一致，更加入了以卷積神經網路來分析使用者的生物習慣，從中分析比對出真正的合法使用者，本方法能在密碼完全外洩或被盜取的狀況下，依然保障著使用者的帳戶，使密碼的安全性更提升一層，本方案的流程圖如 Figure 3.1 所示。

3.1 生物特徵擷取

所謂的生物特徵就是使用者的密碼輸入習慣，每個人在打字時的習慣都不盡相同，特別是在輸入密碼這種經常使用的特殊用詞時，使用者往往都已經養成了某種特殊習慣，例如某兩個按鍵常常一起按下，或特別慣用左右 shift。因此在這階段，我們側錄了使用者打密碼時的鍵盤動作，其中不只包含按鍵代碼，還 包含了按鍵的下壓時間和離開時間，透過這些時間，就能分析出按鍵的順序和間隔。之所以要取得按鍵代碼是因為鍵盤上每個鍵的代碼都不同，但有可能打出相同的結果，例如數字鍵盤和左右 shift，取得鍵盤代碼可更有效且明顯的區分出合法使用者和非法使用者的生物特徵差異。

3.2 生物特徵分析

本論文使用卷積神經網路作為機器學習的模型，由於卷積神經網路有著強大的特徵關聯能力，更適合分析這種差異細微的問題。我們的輸入特徵共有 64 個，分成六個區段，詳細如下 Table 3.1：

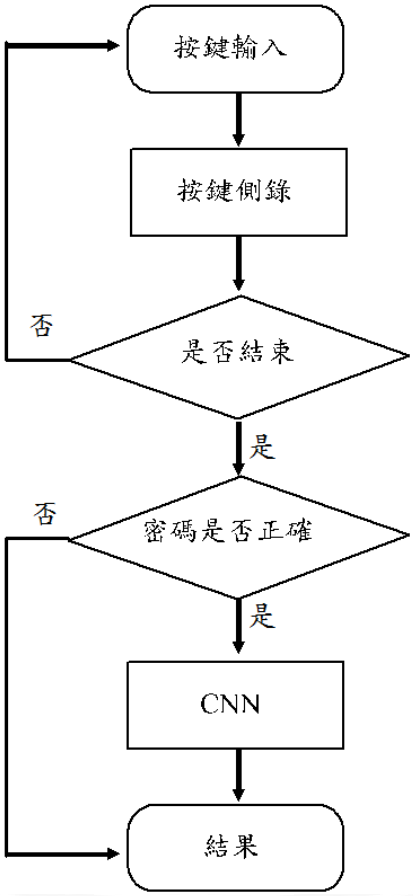

Figure 3.1: 所提方案之流程圖

Table 3.1: CONTENT OF INPUT ARRAY FEATURES

Segment	Features	Length
a	第 1～10 碼的鍵盤代碼	10
b	第 1～10 碼的下壓時間	10
c	第 1～10 碼的放開時間	10
d	第 1～10 碼是否和正確密碼相同	10
e	補 0	15
f	和前一碼的時間差	9

我們把輸入的特徵分成 6 個種類，並且分別將這 6 個種類的數值依照 Figure 3.2 表示的順序放入 8*8 的二為陣列內，如此分部資料可讓卷積層在提取資料時依序比較每個按鍵之間的關聯性，如 Figure 3.2 中的紅色區域為卷積層第一次提取，此時獲得第一個按鍵和第三個按鍵的特徵，並讓這兩個按鍵產生關聯，接著提取藍色區域，這時提取的特徵就具有第一個按鍵到第四個按鍵的關聯，以此類推，直到綠色區域時，第三碼的按鍵又和第五碼的特徵產生關聯，如此重複執行後，便可將按鍵與按鍵互相關聯在一起，產生出新的特徵，由於這次卷積層中無法將所有按鍵互相關聯，因此我們採用兩個卷積層，卷積層的提取範圍互不相同，使特徵關聯度盡可能最大化，在輸入的特徵中，含有部分補 0 的特徵

值，這是為了考量未來有可能擴張密碼長度，先預留的空間，且機器學習時，電腦會發現這些補 0 的特徵無論權重如何改動都不影響結果，因此會自動忽略這些特徵，不會影響結果。

a[0]	b[0]	c[0]	d[0]	a[1]	b[1]	c[1]	d[1]
a[2]	b[2]	c[2]	d[2]	a[3]	b[3]	c[3]	d[3]
a[4]	b[4]	c[4]	d[4]	a[5]	b[5]	c[5]	d[5]
a[6]	b[6]	c[6]	d[6]	a[7]	b[7]	c[7]	d[7]
a[8]	b[8]	c[8]	d[8]	a[9]	b[9]	c[9]	d[9]
f[0]	f[1]	f[2]	f[3]	f[4]	f[5]	f[6]	f[7]
f[8]	e[0]	e[1]	e[2]	e[3]	e[4]	e[5]	e[6]
e[7]	e[8]	e[9]	e[10]	e[11]	e[12]	e[13]	e[14]

Figure 3.2: 所提方案輸入資料存放順序

本論文採用反向傳遞的類神經網路，並且為監督式學習，詳細運作流程如下

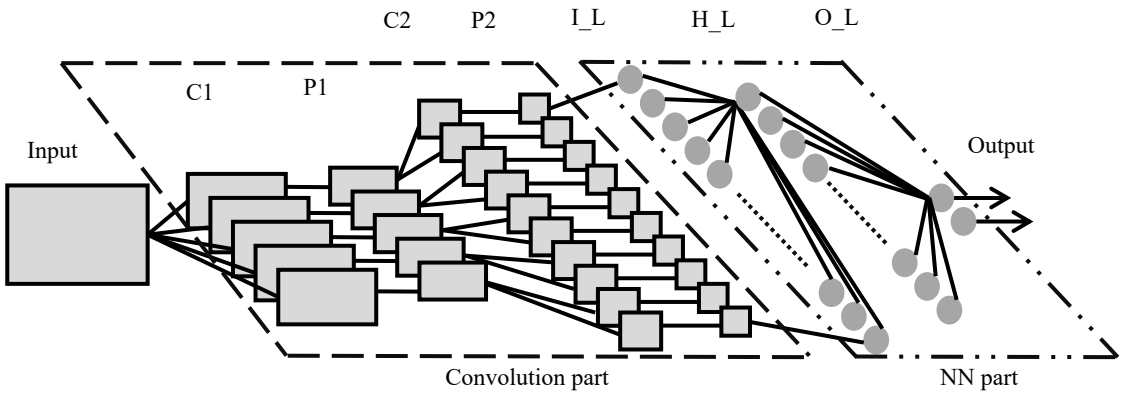

Figure 3.3: 卷積神經網路架構圖

Figure 3.3：為本論文所採用的卷積神經網路架構圖，分為 A ~ H 共 8 個階段：

3.2.1 階段 A

本階段是資料輸入，輸入的內容是生物特徵所擷取的特徵陣列，陣列大小為 64，本模型會將輸入陣列轉換為 8*8 的形式作分析，如此可依照上述方法設計最大化特徵的關聯性。

3.2.2 階段 B

本層為卷積層，Figure 3.3 中的 C1，是本模型中的第一個卷積層，由輸入層提取特徵而來，提取的範圍是 2*4 的二維矩陣，厚度轉換為 32，2*4 的提取可讓局部特徵關聯在一起，再配合卷積單元的順向移動提取，可同時整理順項特徵，激勵函數使用 ReLU，這個激勵函數可讓小於 0 的值轉換為 0，大於 0 的值保持 不變，以確保輸出值不會有負數。

3.2.3 階段 C

本層為池化層，Figure 3.3 中的 P1，是本模型中的第一個池化層，池化的範圍是 2*2 的矩陣，跨步(stride)為 1，如此可避免跨步過大導致訊息大量流失，跨步過大是指選取範圍過大或一次移動距離太遠，這樣會讓池化層在整理特徵時，前者會使特徵平均化，後者會跳過太多特徵，最終導致誤判。

3.2.4 階段 D

本層為卷積層，Figure 3.3 中的 C2，是本模型中的第二個卷積層，本階段和 B 階段大致相似，不同的是本階段的特徵提取範圍是4*2，並且高度變換為 64，利用高度增加再次擴充特徵的多樣性。

3.2.5 階段 E

本層為池化層，Figure 3.3 中的P2，是本模型中的第二個池化層，執行內容與階段 C 一致。

3.2.6 階段 F

本層為輸入層，Figure 3.3 中的 I_L ，是本模型中，類神經網路的輸入層，輸入 的值為階段 E 運算後的結果，是一個二維矩陣，本階段使用的激勵函數依然是 ReLU。值得注意的是，在本階段中加入了 dropout 的功能，該功能可使機器學習忽略掉一部分的神經元權重，來避免過度擬合 (overfitting) 的問題，過度擬合 (overfitting) 是機器學習中一個很常見的問題，發生的原因主要是機器過度訓練，導致機器的學習太過於吻合訓練資料，這會造成機器在分辨新資料時更容易誤判，解決過度擬合的方法有很多，這裡使用的是

忽略權重，忽略掉的權重比例為 30%，輸出的陣列大小為 128。

Figure 3.4 過度擬合的示意圖，圖中黑、白圓點代表不同類型資料，預期分析以一條線區格出本圖的趨勢，黑實線為預期答案，但由於機器的過度訓練，導致預測的曲線完全區分了所有資料，若此時有新的資料加入就很有可能導致誤判，這種狀況就是過度擬合，圖中黑虛線所示。

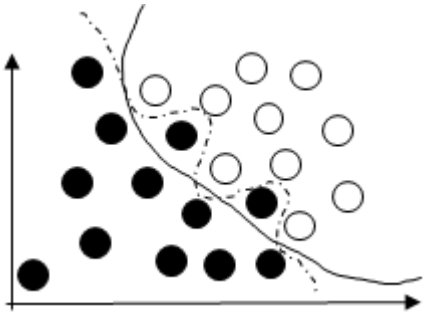

Figure 3.4: overfitting 示意圖

3.2.7 階段 G

Figure 3.3 中的 H_L，是本模型中，類神經網路的隱藏層，神經元的數量為 128，輸出陣列大小為 2，輸出時，帶入激勵函數 softmax，這個函數的主要功 能是將值全部轉換為 0 ～ 1 之間，如此可讓下一階段的分類作標準。

3.2.8 階段 H

這是本模型的最後一個階段，也是輸出層，在這階段前，機器學習會先篩選輸入的結果，最大值會轉換為 1，其餘的全部歸零，透過這功能達到分類效果，也因此可以區分出合法使用者和非法使用者。

本論文所提出的方法，資料在進入類神經網路前就先經過整理，因此在階段 B 時可以串聯多個按鍵的代碼特徵，如此讓這些特徵具備關聯，再透過池化層的特徵過濾，可大幅增加預測的準確度。為了將準確度提升到最高，本論文做了幾種實驗，同時調整不同的參數，以求完美，詳細內容請看實驗分析與討論。

3.3 GPU 平行設計

由於類神經網路中具有大量神經元，神經元又有權重和偏量，這些在程式內都以陣列存放，因此在運算時，可以將每個神經元交由 GPU 上的串流處理器運算，其中偏量和權重放在全域記憶體內，讀入的特徵資料放入共享記憶體，激勵函數由常數記憶體在一開始時就先定義，如此一來在運算時，每個串流記憶體便能獨立運算，減少發生碰撞和傳輸，可再次提升效能，詳細配置如 Figure 3.5。

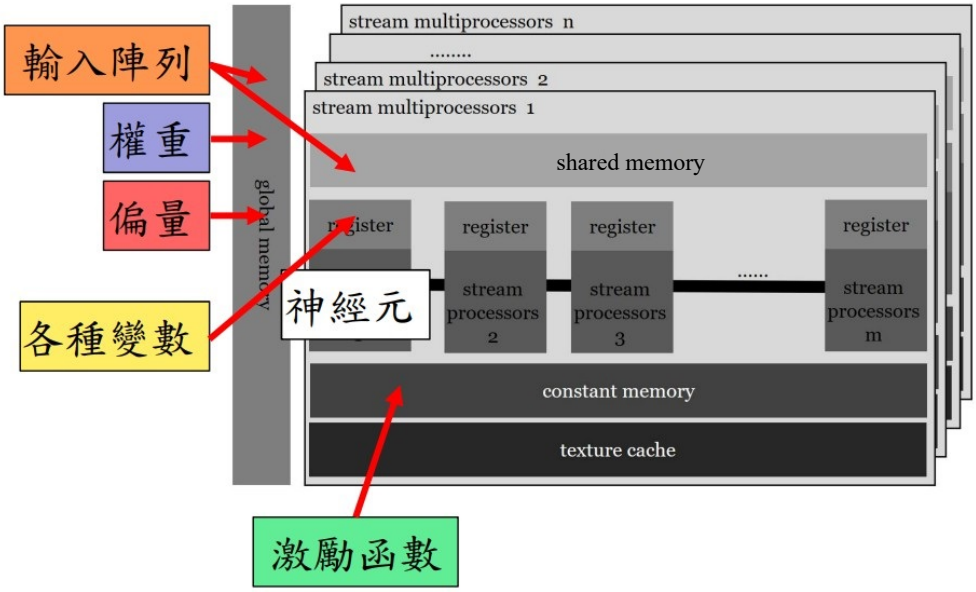

Figure 3.5: GPU 平行化設計的變數分配

第四章　實驗分析與討論

4.1 實驗介紹

本論文提出了一個以擊鍵習慣為特徵的身份驗證法，非法使用者即使取得了完整密碼，依然難以登入帳戶，為了驗證所提出方法的可靠度，我們設計了以下三種實驗，NN 和 CNN 的分析能力比較、特徵可靠度分析和 GPU 效能優化，實驗的環境如下：

Table 4.1: 實驗環境

cpu	Intel Q8200 2.33GHz*4
memory	3.9GB
os	Ubuntu 16.04 64-bit
gpu	Nvidia GTX 650Ti 2GB memory 768 CUDA core
機器學習工具	Google tensorflow

本方案所採用的機器學習為監督式學習 (supervised)，因此必須具備練習資料，實驗中我們採用的資料分別向 18 位不同的使用者側錄取得，分別為 A 到 R，由於本論文立論於密碼已被他人知道，因此在側錄前，就會將 A 習慣使用 的密碼告訴 B 到 R 使用者，共 17 人，接著 18 位受測者每人輸入 100 次密碼，並且從每人中隨機挑取 10 筆當作測試資料，其餘當作訓練資料，因此訓練資料有 1620 筆，測試資料有 180 筆。本實驗採用錯誤接受率 (False Acceptance Rate, FAR) 和錯誤拒絕率 (False Rejection Rate, FRR) 來當作評價標準，其中錯誤接受率代表非法使用者成功登入的機率；錯誤拒絕率代表合法使用者無法成功登入的機率。

$$FAR = \frac{\text{非法使用者成功登入的次數}}{\text{全部測試的次數}} \quad (3)$$

$$FRR = \frac{\text{合法使用者被拒絕登入的次數}}{\text{全部測試的次數}} \quad (4)$$

4.2 類神經網路

本次實驗中測試了傳統類神經網路的辨識率，訓練資料如上述所說，訓練次數由5000 到25000 次，其結果如下Table 4.2：

Table 4.2: 本方案利用 NN 的辨識率

訓練次數 (次)	NN 準確率 (%)	FRR(%)
5000	94.44	100
7000	94.44	100
9000	94.44	100
11000	94.44	100
13000	94.44	100
15000	94.44	100
17000	94.44	100
19000	94.44	100
21000	94.44	100
23000	94.44	100
25000	94.44	100

由本次實驗結果可以看出，雖然辨識的準確率很高，但是合法使用者被誤判的可能性是100%，並且無論如何學習，依然無法改變誤判的事實，因此可以得知本實驗無法採用傳統神經網路來分析。

4.3 卷積神經網路

本次實驗改變了機器學習的模型，改採用卷積神經網路，藉由神經網路的特性，大幅改善了學習的結果，詳細如下 Table 4.3。

透過卷積層的區域特徵過濾，確實提升了機器學習的辨識率，並且在 25000 次學習後，準確率達到 99%，非法使用者 100% 的阻擋了，合法使用者也只有 1 次誤判，是相當明顯的改善，因此本論文選用卷積神經網路作為分析工具。

Table 4.3: 本方案利用 CNN 的辨識率

訓練次數 (次)	CNN 準確率 (%)	FRR(%)
5000	94.44	100
7000	94.44	100
9000	95.56	80
11000	97.22	50
13000	97.22	50
15000	98.33	30
17000	98.33	30
19000	98.89	20
21000	98.89	20
23000	98.89	20
25000	99.44	10

4.4 特徵可靠度分析

為了證明生物習慣具有明顯的特徵，並足以區分使用者，因此本論文設計了一個實驗，實驗內設定的合法使用者分別為 A～R 輪流擔任，當 A 是合法使用者時，B～R 設定為非法使用者，讓機器分辨出使用者 A；當 B 為合法使用者時，A 和 C～R 設定為非法使用者，讓機器分辨出使用者 B，以此類推，在 25000 次訓練後的詳細結果如下 Table 4.4：

Table 4.4: 特徵可靠度分析結果

合法使用者	準確率 (%)	FAR(%)	FRR(%)
A	99.44	0	10
B	99.44	0	10
C	98.33	0	30
D	99.44	0	10
E	99.44	0	10
F	98.33	0	30
G	98.89	0	20
H	99.44	0	10
I	98.89	0	20
J	98.89	0	20
K	99.44	0	10
L	99.44	0	10
M	98.89	0	20
N	99.44	0	10
O	99.44	0	10
P	99.44	0	10
Q	99.44	0	10
R	99.44	0	10

由本實驗可見所有的使用者的辨識率都超過了 97%，FAR 更是全部為 0%，

代表沒任何的非法使用者能成功登入，缺點是有部分使用者容易被誤判為非法使用者，誤判率達到了30%，這個原因是訓練資料不均衡造成的，為了改善，本論文設計了以下這個實驗。

4.5 不均衡資料分析

訓練資料不均等在機器學習中是一個很常見的問題，這會造成機器的學習偏差導致結果嚴重誤判，例如在訓練資料中，有 90% 的資料屬於類型 1，有 10% 資料屬於類型 2，機器在幾次學習後就會發現只要他猜測類型 1，就會有高達 90% 的準確率，因此之後機器不管看到什麼資料，都先猜類型 1，導致整個機器學習無法正確分類。在本論文中，之前的學習資料確實有著明顯不均衡的學習資料，非法使用者的資料遠多於合法使用者，為了改善這個問題，本論文將訓練資料改為合法使用者 1530 筆，非法使用者 1530 筆 (每人 90 筆，共 17 人)，測試資料改成合法使用者 170 筆，非法使用者 170 筆 (每人 10 筆，共 17 人，資料皆與訓練資料不同)，測試 5 次，其結果如下 Table 4.5：

Table 4.5: 不均等資料分析結果

FAR(%)	FRR(%)	準確率 (%)
0	8.82(15 筆)	95.59
0	8.82(15 筆)	95.59
0	9.41(16 筆)	95.29
0	8.82(15 筆)	95.59
0	9.82(16 筆)	95.59

由實驗結果看出合法使用者被誤判的機會大幅下降了，非法使用者依然無法登入，沒有一次例外，並且整體的準確率達到驚人的 95%。合法使用者被拒絕的機率僅有 8%。若合法使用者不幸遭到本程式誤判，僅需要再輸入一次就可以，連續兩次遭到誤判的機率不到 0.7%，可以說是微乎其微；非法使用者無論輸入幾次，阻擋率都高達 100%，完全無法登入，證明本方法確實大幅增加密碼的可靠度。

4.6 貧乏資料分析

這是監督式機器學習面臨的一個問題，也是相當符合現實情況的問題，在實際 使用上，非法使用者的學習資料通常是難以取得的，因此一個好的身分驗證法必須要在無學習資料的情況下分辨出合法使用者。為了印證此一問題，本次實驗中，訓練資料內僅有 90 筆非法使用者和 100 筆合法使用者，非法使用者的資料出自於同一個使用者，測試資料為 270 筆，分別由 100 筆合法使用者 A 和 170 筆非法使用者 B～R 組成(17 人，每人 10 筆)，其中

A 和 B 在有部分資料在訓練資料內，因此有 16 個使用者是機器完全沒看過的，機器必須從這些人中區分出非法使用者，測試本模型對於無資料的非法使用者的抵抗能力，其結果如下 Table 4.6：

Table 4.6: 貧乏資料分析結果

FAR(%)	FRR(%)	準確率 (%)
29.41(50 筆)	14	76.29

由實驗結果可見誤判率上升了，準確率也下降了很多，可是對於非法使用者依然阻擋了 72% 左右，使帳號的可靠度增加了 72%，整體來說準確率也達到 75%。雖然看起來並沒有之前實驗的那麼優秀，但是隨著收集資料越來越多，就越能改善這個問題，本次實驗是在最少的學習資料下產生的結果，未來實際使用時的學習資料絕對會大於此，也可以說是本實驗為最低準確度，因此本方法在實際運用上有不錯的表現。

4.7 GPU 平行

本論文考量到未來在應用上會有大量訓練資料和大量的訓練次數，因此加速本方案是必備的，本論文採用 GPU 平行化運算整個卷積神經網路模型，並且達到超過 4 倍的加速效能，加速比公式如 Eq.(5)

$$\text{加速比}=\frac{\text{CPU 執行時間}}{\text{GPU 執行時間}} \tag{5}$$

詳細結果如下 Table 4.7：

Table 4.7: 本方案在 CPU 和 GPU 上執行的比較

訓練次數 (次)	CPU(秒)	GPU(秒)	加速比 (倍)
1000	108.226	37.631	2.88
3000	302.721	102.17	2.96
5000	629.372	196.442	3.20
7000	1037.256	281.032	3.69
9000	1355.373	343.153	3.95
11000	1601.097	402.388	3.98
13000	1935.493	477.17	4.01
15000	2195.907	532.881	4.12
17000	2519.198	624.386	4.03
19000	2984.384	686.782	4.35
21000	3300.136	747.132	4.41
23000	3663.377	791.124	4.63
25000	4096.983	833.672	4.91

由 Figure4.1 可以看出 CPU 的運算時間隨著訓練次數增加而大幅成長，但 GPU 的運算時間增加不多。當訓練 17000 次時，能達到 4 倍加速，訓練超過

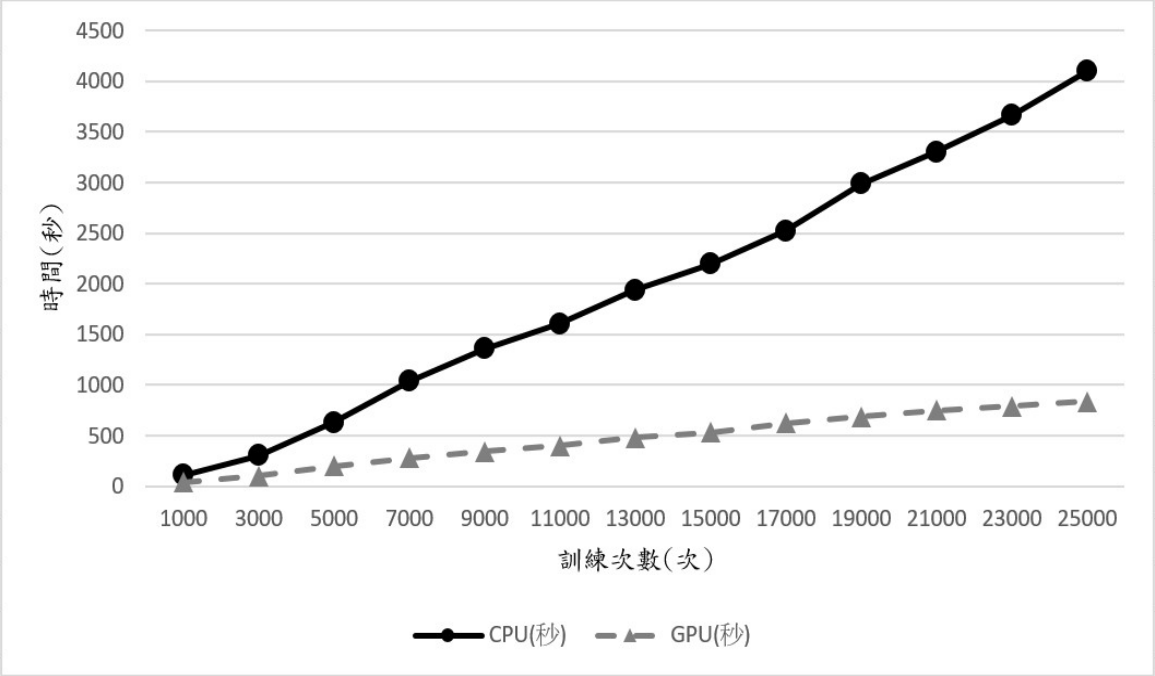

Figure 4.1: GPU 和 CPU 的運算時間成長曲線圖

25000 次的時候更是近 5 倍，在日後的實際應用上，若有更龐大的訓練資料或更多的訓練次數，將會有更高的加速比，對於應用會有卓越的幫助。

4.8 討論

本論文提出一種身分驗證方法，不僅只比對密碼的正確性，更驗證了使用者的生物習慣，並且在這細微差異中找出了真正的合法使用者。從實驗 4.2 可以看出，即便是類神經網路都無法分析預測出生物習慣，本論文採用卷積神經網路，透過卷積層和池化層的特徵提取能力，有效找出了生物擊鍵特徵中的差異，再配合一些參數的調整與修正，使整個神經網路的分析預測能力達到 96% 以上，其中非法使用者成功登入的機會趨近於零，大幅降低了帳戶遭竊的風險。本論文也假設了各種在實際使用上的情況，使整個方法更實用，對於未來推廣更有幫助。

第五章　結論

隨著科技快速發展，密碼和人們的關係已經密不可分，但科技發展的同時也加速了密碼被破解的速度，使密碼不再安全，許多人為了增加密碼的可靠度，不斷增加密碼的長度和複雜度。為了不遺忘太過複雜的密碼，人們常會把密碼記下來，不論是手寫或是檔案的形式，都增加了密碼外洩的風險，為了解決這個問題，並且在不增加密碼長度和複雜度的前提下，本論文提出了一種身份辨識法，在即使是整個密碼已經外洩的情況下，依然有著高度的保護，有效杜絕非法使用者登入。

本論文提出一個方法，利用人們打字的習慣當作特徵，再透過機器學習來分析，其中根據實驗 4.2 和實驗 4.3 的比較可以得知，以卷積神經網路為模型的機器學習比傳統類神經網路更具有分析和辨識能力。並且從表 4.3 中可以看到，機器學習在 25000 次訓練後，可以達到 99% 的準確率，僅有一筆合法使用者遭 到誤判為非法使用者，非法使用者被完全拒絕登入，顯示本方法有著相當卓越 的身份辨識能力。本論文更針對生物習慣是否具有身份區別特徵來做分析，由實驗 4.4 可以看出生物習慣確實能有效分辨出各個使用者，證明生物習慣能當作身份驗證的機制。在後面的實驗中，調整了一些實驗的參數，透過實驗 4.5 更是將準確率提升到了 95% 以上，辨識能力相當卓越。本論文也考量在實際使用中會缺乏數據，也設計了缺乏數據的實驗 4.6，由實驗結果顯示，即使不具備訓練資料，本方法依然能阻擋高達 71% 的非法使用者。雖然有 29% 左右的非法使用者能成功登入，但是本論文是假設完整密碼外洩為前提做的實驗，非法使用者在擁有完整密碼的情況下被阻擋，多增加了密碼 71% 的可靠度，確實明顯的改善了密碼不安全的問題。最後本論文加入 GPU 平行化運算，加速訓練的效能，日後使用時如果有龐大的訓練資料或有極高的訓練次數，將會有卓越的幫助。

本論文提出了一種身份驗證方式，能在密碼完整曝光的情況下，阻擋接近所有的非法使用者，並且僅有少部分的合法使用者會被誤判阻擋，未來將針對這些誤判做研究，變換機器學習模型或修改參數，以優化整體的準確率，並且收集更多的使用者數據做測試，來證明本方法的價值。

References

[1] Hagan, Martin. Neural Network Design. PWS Publishing Company. 1996. ISBN 7-111-10841-8.

[2] F. Monrose and A. D. Rubin, “Keystroke dynamics as a biometric for authentication,” Future Gener. Comput. Syst., vol. 16, no. 4, pp. 351–359, Feb. 2000.

[3] Attila Ceffer, Janos Levendovszky, “Kolmogorov-Smirnov test for keystroke dynamics based user authentication” Computational Intelligence and Informatics (CINTI), 2016.

[4] Y. Zhang, G. Chang, L. Liu, and J. Jia, “Authenticating user’s keystroke based on statistical models,” in Genetic and Evolutionary Computing (ICGEC), 2010 Fourth International Conference on, pp. 578–581, Dec 2010.

[5] K. Buza and D. Neubrandt, “How you type is who you are,” in 11th IEEE International Symposium on Applied Computational Intelligence and Informatics, 2016.

[6] P. Chairunnanda, N. Pham, and U. Hengartner, “Privacy: gone with the typing! identifying web users by their typing patterns,” in Privacy, Security, Risk and Trust (PASSAT) and 2011 IEEE Third Inernational Conference on Social Computing (SocialCom), 2011.

[7] P. Kang and S. Cho, “Keystroke dynamics-based user authentication using long and free text strings from various input devices,” Inf. Sci., vol. 308, no. C, pp. 72–93, Jul, 2015.

[8] E. Alpaydin, Introduction to Machine Learning, 2nd ed. The MIT Press, 2010.

[9] A. B. J. T. Pin Shen Teh and S. Yue, "A survey of keystroke dynamics biometrics," The Scientific World Journal, vol. 2013.

[10] Joe Lemley, Shabab Bazrafkan, Peter Corcoran, "Deep learning for consumer devices and services: pushing the limits for machine learning, artificial intelligence, and computer vision" IEEE Consumer Electronics Society, 2017.

[11] Panos Louridas, Christof Ebert, Machine Learning, IEEE Computer Society, pp.110-115, 2016.

[12] Convolutional Neural Networks (LeNet) - DeepLearning 0.1 documentation. DeepLearning 0.1. LISA Lab, 2013.

[13] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," Journal of Machine Learning Research, vol. 15, no. 1, pp. 1929–1958, 2014.

[14] A. Krizhevsky, I. Sutskever, and G. Hinton. Imagenet classification with deep convolutional neural networks. In NIPS, 2012.

[15] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In ICML, 2015.

[16] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In CVPR, 2015.

[17] google tensorflow. https://www.tensorflow.org/

[18] nVidia CUDA. http://www.nvidia.com.tw/object/cuda- parallel-computing-platform-tw.html