



朝陽科技大學
資訊與通訊系

碩士論文

基於OpenPose骨架及LSTM/GRU模型之跌倒檢測研究

Research on Fall Detection Based on OpenPose Skeleton and
LSTM/GRU Models

指導教授：林傳筆 博士

研究生：管維凱

中華民國 109 年 7 月

The logo of Chaoyang University of Technology is a circular emblem. It features a central sunburst design with rays emanating from a central point. The text "CHAORYANG UNIVERSITY OF TECHNOLOGY" is written in a circular path around the top half of the emblem, and "朝陽科技大學" is written around the bottom half. The year "1994" is positioned at the bottom center of the emblem.

朝陽科技大學資訊與通訊系

Department of Information and Communication

Engineering

Chaoyang University of Technology

碩士論文

Thesis for the Degree of Master

基於 OpenPose 骨架及 LSTM/GRU 模型之跌倒檢測研究

Research on Fall Detection Based on OpenPose Skeleton and

LSTM/GRU Models

指導教授：林傳筆 博士 (Chuan-Bi Lin)

研究生：管維凱 (Wei-Kai Kuan)

中華民國 109 年 7 月

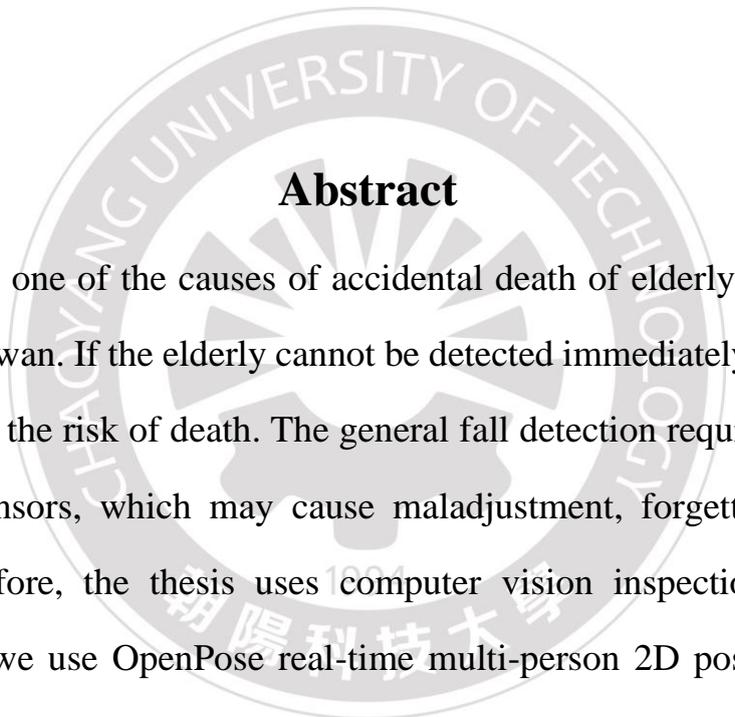
July, 2020



摘要

跌倒是台灣65歲以上年長者事故傷害死亡的原因之一，當年長者發生意外時，若無法即時被察覺，將會增加死亡風險。常見的跌倒檢測需要檢測者穿戴感測器，可能造成適應不良、忘記穿戴或遺失。因此，本文使用電腦視覺檢測，不須要感測器，用OpenPose即時多人2D姿勢估計來抓取人體的25個關節點，透過關節點的軌跡變化就能辨識出人體的動作。除了能有效抓到人體關節點，在嚴苛的空間環境下有不錯的效果，因為使用關節點取代影像，能有效改善訓練時間過長、影像模糊及光影變化等問題。本文使用單一視角影像，來降低設備成本，搭配有時間序列的遞歸神經網路(RNN)、長短期記憶(LSTM)和門控循環單元(GRU)進行實驗，來對連續時間的人體關節點變化進行學習。在實驗結果中，LSTM檢測率達98.2%和GRU檢測率為97.3%，因此本文提出跌倒檢測法之準確率相當高。

關鍵詞：OpenPose、2D姿勢估計、遞歸神經網路、長短期記憶、門控循環單元、跌倒偵測、動作識別。



Abstract

Falling is one of the causes of accidental death of elderly people over 65 years old in Taiwan. If the elderly cannot be detected immediately in an accident, he will increase the risk of death. The general fall detection requires the detector to wear the sensors, which may cause maladjustment, forgetting to wear or missing. Therefore, the thesis uses computer vision inspection without any sensors. Here, we use OpenPose real-time multi-person 2D pose estimation to grab 25 joint points of the human body. The movement of the joint point is recognized through the movement of the human body. In addition to being able to effectively grasp the joints of the human body, it has a good effect in the harsh space environment. Because the use of joint points instead of images, the training time is effectively improved. In addition, it also improves image blur, light, and shadow changes. The thesis uses single-view images to reduce equipment costs. We experimented with time series recurrent neural network (RNN), long and short-term memory (LSTM), and gated recurrent unit (GRU). The model learns the changes of human joint points in continuous time. In the experimental results, the LSTM is 98.2% and the GRU is 97.3%. Therefore, the accuracy of the fall detection method proposed in this article is quite high.

Keywords: OpenPose, 2D Pose Estimation, Recurrent Neural Network, Long Short-Term Memory, Gated Recurrent Units, Fall Detection, Action Recognition.

The logo of Chaoyang University of Technology is a circular emblem. It features a central sunburst design with rays emanating from a central point. The text "CHAORYANG UNIVERSITY OF TECHNOLOGY" is written in a circular path around the top half of the emblem, and "朝陽科技大學" is written around the bottom half. The year "1994" is positioned at the bottom center of the emblem.

致謝

在朝陽科技大學就讀研究所的這段時間裡，受到了許多人的協助，協助我課業的專業知識、處理公務以及論文的撰寫等，同時也學習到更多待人處事的訣竅，非常感謝這一路上與我相遇的這些人，給我的支持與照顧。

首先，感謝指導教授林傳筆老師，從大學的班導師到碩士班的指導教授，無論在哪個學習歷程上，都幫助我許多事情以及鼓勵，指引我未來正確的方向。同時，感謝幫忙指導的兩位教授黃永發老師和陳忠信老師，協助我論文的研究以及撰寫，在求學的路上受益良多。感謝育強學長、建民同學、俊邑同學、懷逸同學、世鈞同學和育陞同學，在遇到困難或問題時總能認真協助，也特別感謝王能中教授及鄭佳炘教授，感謝二位百忙之中前來擔任口試委員，感謝各位的幫忙及陪伴，讓我在碩士的生涯留下美好的回憶。

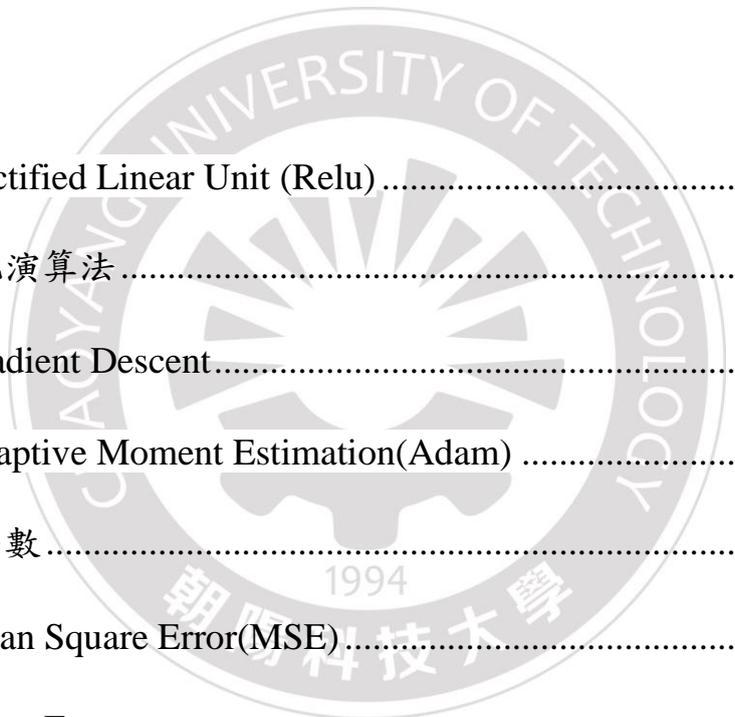
最後，要特別感謝我的家人，盡全力支持我繼續讀碩士，並適時地給予我鼓勵與關懷，由衷地感謝一路上陪著我成長的每個人。

管維凱 僅誌



目錄

摘要.....	I
Abstract.....	II
致謝.....	III
目錄.....	IV
圖目錄.....	VIII
表目錄.....	X
第一章 緒論.....	1
1.1 研究背景.....	1
1.2 研究動機與目的.....	3
1.3 論文章節介紹.....	4
第二章 相關研究.....	5
2.1 人工神經網路.....	5
2.1.1 多層感知器(MLP).....	5
2.1.2 卷積神經網路(CNN).....	6
2.1.3 遞歸神經網路(RNN).....	8
2.1.4 長短期記憶 (LSTM).....	9
2.1.5 門控遞歸單元(GRU).....	11
2.2 激勵函數.....	12
2.2.1 Logistic Sigmoid (Sigmoid).....	12
2.2.2 Hyperbolic Tangent Sigmoid (Tanh).....	12



2.2.3 Rectified Linear Unit (Relu)	13
2.3 最佳化演算法	13
2.3.1 Gradient Descent.....	13
2.3.2 Adaptive Moment Estimation(Adam)	13
2.4 損失函數	14
2.4.1 Mean Square Error(MSE).....	14
2.4.2 Cross Entropy	15
2.5 模型評估指標	15
2.5.1 靈敏度(Sensitivity).....	16
2.5.2 特異度(Specificity).....	17
2.5.3 準確度(Accuracy).....	17
2.6 OpenPose提取骨架.....	17
2.7 正規化(Normalization)	18
2.7.1 Z分數標準化(Z-Score Standardization)	18
2.7.2 最小值最大值正規化(Min-Max Normalization)	18
2.8 插值方法	18
2.8.1 線性插值.....	19
2.9 文獻探討	19
2.9.1 穿戴式裝置跌倒檢測.....	19

2.9.2 環境感測裝置跌倒檢測.....	20
2.9.3 電腦視覺跌倒檢測.....	20
第三章 基於OpenPose骨架的LSTM/GRU模型.....	22
3.1 模型訓練流程.....	22
3.2 數據集.....	23
3.2.1 UR Fall Detection Dataset.....	23
3.2.2 Fall Detection Dataset.....	24
3.3 OpenPose取得人體骨架.....	25
3.4 數據預先處理.....	25
3.4.1 格式調整.....	27
3.4.2 相對位置正規化.....	27
3.4.3 補償遺失值.....	30
3.5 模型架構.....	34
3.6 實驗環境.....	36
3.6.1 資料處理環境.....	36
3.6.2 訓練環境.....	36
第四章 實驗結果.....	38
4.1 RNN、LSTM及GRU模型訓練比較.....	38
4.1.1 正規化方法比較.....	38
4.1.2 隱藏神經元比較.....	45



4.1.3 學習率比較.....	49
4.2 RNN、LSTM及GRU模型評估指標.....	51
4.3 RNN、LSTM及GRU模型訓練速度比較.....	56
4.3.1 硬體比較.....	56
4.3.2 正規化方法比較.....	57
4.4 小結.....	57
第五章 結論與未來工作.....	59
參考文獻.....	60



圖目錄

圖 2-1、多層感知機(MLP).....	5
圖 2-2、卷積神經網路(CNN)	6
圖 2-3、卷積運算.....	7
圖 2-4、最大池化運算.....	7
圖 2-5、遞歸神經網路(RNN)	8
圖 2-6、LSTM	9
圖 2-7、GRU	11
圖 2-8、混淆矩陣.....	16
圖 3-1、模型訓練流程.....	22
圖 3-2、UR Fall Detection Dataset設備位置	23
圖 3-3、Fall Detection Dataset設備位置.....	24
圖 3-4、人體骨架關節點(25點).....	26
圖 3-5、OpenPose抓取到的關節點位	26
圖 3-6、原始的人體骨架座標位置及數量	29
圖 3-7、位移後的關節點位置及數量	29
圖 3-8、關節點遺失.....	30
圖 3-9、手部y點變化.....	31
圖 3-10、線性插值補償遺失點(第15點遺失).....	32
圖 3-11、線性插值補償遺失點(第11至19點遺失).....	32
圖 3-12、線性插值補償遺失點(第10至14點, 第16至20點遺失).....	33
圖 3-13、跌倒模型架構(輸入50).....	35
圖 3-14、跌倒模型架構(輸入30).....	35

圖 4-1、RNN模型訓練及驗證準確度(a)訓練準確度 (b)驗證準確度.....	39
圖 4-2、RNN模型訓練及驗證損失(a)訓練損失 (b)驗證損失.....	40
圖 4-3、LSTM模型訓練及驗證準確度(a)訓練準確度 (b)驗證準確度	41
圖 4-4、LSTM模型訓練及驗證損失(a)訓練損失 (b)驗證損失	42
圖 4-5、GRU模型訓練及驗證準確度(a)訓練準確度 (b)驗證準確度.....	43
圖 4-6、GRU模型訓練及驗證損失(a)訓練損失 (b)驗證損失.....	44
圖 4-7、不同正規化方法平均準確度比較	45
圖 4-8、RNN不同神經元數量之收斂速度比較.....	46
圖 4-9、LSTM不同神經元數量之收斂速度比較.....	48
圖 4-10、GRU不同神經元數量之收斂速度比較.....	49
圖 4-11、不同學習率之平均準確度比較(相對位置).....	50
圖 4-12、不同學習率之平均準確度比較(線性插值).....	50
圖 4-13、RNN混淆矩陣(a)未正規化 (b)最大最小值 (c)相對位置	52
圖 4-14、RNN模型正規化方法評估指標比較.....	52
圖 4-15、LSTM混淆矩陣(a)未正規化 (b)最大最小值 (c)相對位置	53
圖 4-16、LSTM模型正規化方法評估指標比較.....	54
圖 4-17、GRU混淆矩陣(a)未正規化 (b)最大最小值 (c)相對位置	55
圖 4-18、GRU模型正規化方法評估指標比較.....	55
圖 4-19、模型訓練速度.....	56
圖 4-20、正規化方法訓練速度比較	57



表目錄

表 3-1、數據集分類數量.....	25
表 3-2、關節點遺失比例.....	34
表 3-3、模型訓練參數.....	37
表 3-4、模型結構參數.....	37
表 4-1、RNN隱藏神經元平均準確度.....	47
表 4-2、LSTM隱藏神經元平均準確度.....	48
表 4-3、GRU隱藏神經元平均準確度.....	49

第一章 緒論

由於高齡化情況逐漸嚴重，獨居年長者或行動不便者也隨之增加。再加上出生率下降等問題，導致扶養比失衡，可能發生照顧者人力不足等情況。跌倒是台灣65歲以上年長者事故傷害死亡的原因之一，當年長者發生意外時，若無法即時被察覺，將會增加死亡風險。因此，我們提出一種使用遞歸神經網路，對於人體骨架姿勢的辨識方法，當跌倒發生時，可以即時發現。

1.1 研究背景

在2018年，我國國家發展委員會的人口推估報告[1]中顯示，至2065年總人口數約減少20%至32%：2018年，總人口約為2,359萬人；至2065年，高、中、低推估之總人口數將分別降為1,880萬人、1,735萬人及1,601萬人。在未來我國的總人口數將會下降，可能導致具生產能力者減少，發生扶養比失衡等問題。

另外，根據《世界人口展望：2019年修訂版》[2]的數據顯示，到2050年，全世界每6人中就有1人年齡在65歲(16%)以上，在2019年為11人(9%)；到2050年時，歐洲和北美每4人中就有1人年齡在65歲或以上。2018年，全球65歲或以上人口史無前例地超過了5歲以下人口數量。除此之外，預計80歲或以上人口將成長兩倍，從2019年的1.43億增至2050年的4.26億。而我國將在未來8年內進入超高齡社會，高齡化速度超過歐美日等先進國家，台灣於1993年邁入高齡化社會



(老年人口占總人口比率超過7%)，並於2018年成為高齡社會(超過14%)，預計將於2026年成為超高齡社會(超過20%)。因醫療水準上升和生育率下降等緣故，導致加速了高齡化情況。同時，高齡化社會將會產生的影響如：醫療資源不足、機構照顧不足和經濟收入減少，支出增加等問題。

而在《世界人口展望：2015年修訂版》[3]的預測，全球生育率將從2010至2015年每名婦女生育2.5個子女，下降到2025至2030年的2.4個，並將繼續下降，在2095至2100年降至2.0個。在2018年，我國出生數預估為18萬至19萬人，因15-49歲育齡婦女人口減少之影響，至2065年，高、中、低推估之出生數將分別再降至13萬人、9萬人及5萬人。依據中推估結果，預估2018-2029學年度，各級平均入學年齡人口將分別減少至20萬人、20萬人及22萬人。社會扶養負擔加重為人口老化後須面臨的一項挑戰。將15-64歲工作年齡人口定義為具生產能力者，而其他年齡者定義為無生產能力者或依賴人口。在2018年每5位生產者負擔1位老年人口，2065年轉變為每1.2位生產者即需負擔1位老年人口。具生產能力者將會供不應求，導致未來年長者扶養比將持續升高，無法照料完善，發生意外的可能性也隨之提升，社會負擔將日益加重。

經由衛生福利部國民健康署的107年死因統計[4]，65歲以上事故傷害死亡原因，第一為交通事故(每十萬人34.7人)，第二為跌倒(每十

萬人25.7人)。世界衛生組織[5]也指出跌傷是世界各地意外或非故意傷害死亡的第二大原因，在致命的跌傷中，65歲以上成年人所佔比例最大，全世界每年估計有64.6萬人因跌傷而死亡。年齡是導致跌傷的主要危險因素之一，老年人因跌倒而死亡或受重傷的風險最大，年齡越大，風險越高。經研究表明，跌倒後即時得到幫助的老年人可以有效地將死亡風險降低80%，將需長期治療的住院風險降低26%。

1.2 研究動機與目的

年長者發生跌倒意外是非常嚴重的問題，若無人及時發現及協助，將會帶來無法挽回的結果。目前關於跌倒檢測的研究，現有的方法可以分為三類。首先，是使用可穿戴傳感器[6][7]來檢測跌倒的發生，利用如三軸加速度計等數據來檢測人的姿勢。第二，使用環境感測器[8]，通過像是振動、紅外線或音頻來檢測跌倒。最後為本文所使用的電腦視覺檢測[9][10]，使用監視攝影機拍攝的影片或圖像序列來檢測跌倒，像是計算身體形狀，姿勢或3D頭部變化分析，透過時空特徵的變化來檢測人體活動姿態。

第一種方法因為需要穿戴傳感器，所以可能發生年長者忘記充電或是配戴，也會發生設備遺失的情況，導致設備成本的增加，甚至有年長者不習慣穿戴式裝置等問題。第二種方法則容易受到環境干擾的問題，也可能需要多個感測器來做輔助而增加成本。本文選擇了使用設備成本較低的方法且不需穿戴傳感器，使用單一攝影設備來實現跌



倒檢測的技術，基於遞歸神經網路對人體骨架的姿勢辨識方法。常見的人體骨架取得方式，如使用Kinect設備[11][12]來取得，但此方法則需要額外的設備成本，因此本文使用OpenPose即時多人2D姿勢估計[13][14]來取得人體骨架。其次，得到的數據需保留時空特徵，因此使用具有時間序列的遞歸神經網路，來學習人體骨架在時間內的變化，得以檢測跌倒的發生。

1.3 論文章節介紹

本論文主要共分為五個章節，各章節之說明如下。第二章節介紹人體姿勢辨識的相關研究；第三章節為介紹系統架構、數據集介紹、預處理方法、模型架構及訓練之流程；第四章節為模型實測結果及分析；第五章節為結論與未來工作。

第二章 相關研究

2.1 人工神經網路

人工神經網路(Artificial Neural Network, ANN)[15-16]，也稱為神經網路或類神經網路。是一種模仿生物神經網路(像是大腦)的結構，與功能的數學模型或計算模型，用於對函式進行估計或近似。主要的特點有三方面：具有學習功能、有聯想儲存功能，與高速找尋最佳解的能力。

2.1.1 多層感知器(MLP)

多層感知機(Multilayer perceptron, MLP)[17-18] 是一種前向結構的人工神經網路，包括至少一個隱藏層(除了輸入層和輸出層以外)。單層感知器只能學習線性函數，而多層感知器可以學習非線性函數。如圖 2-1所示，多層感知器分為三層，輸入層、隱藏層和輸出層。

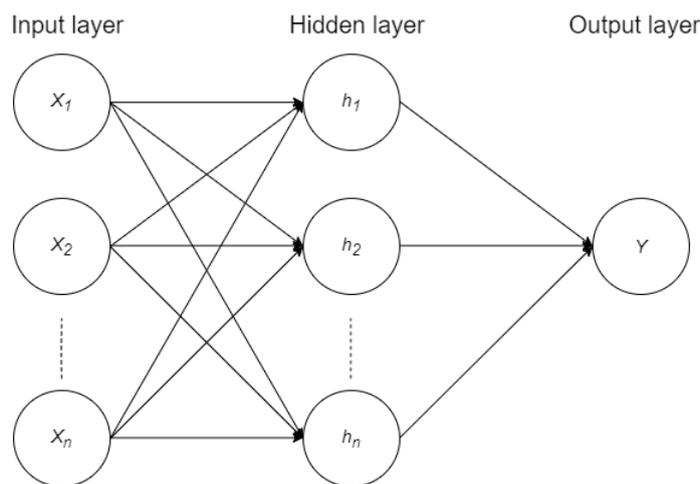


圖 2-1、多層感知機(MLP)[19]

輸入一系列特徵 $X = (x_1, x_2, \dots, x_n)$ 和目標 Y ，多層感知器能以分類或回歸為目的，學習到特徵和目標之間的關聯。再透過反向傳播算法 (Backpropagation, BP)[19]，更新每一神經元的權重和偏權值。

2.1.2 卷積神經網路(CNN)

卷積神經網路(Convolutional Neural Network, CNN)[20-21]是一種前饋神經網路，如圖 2-2所示，CNN由一個或多個卷積層(Convolutional Layers)和全連接層(Fully Connected Layers)通常由ANN或MLP組成，同時也包括關聯權重和最大池化層(Max Pooling Layer)。

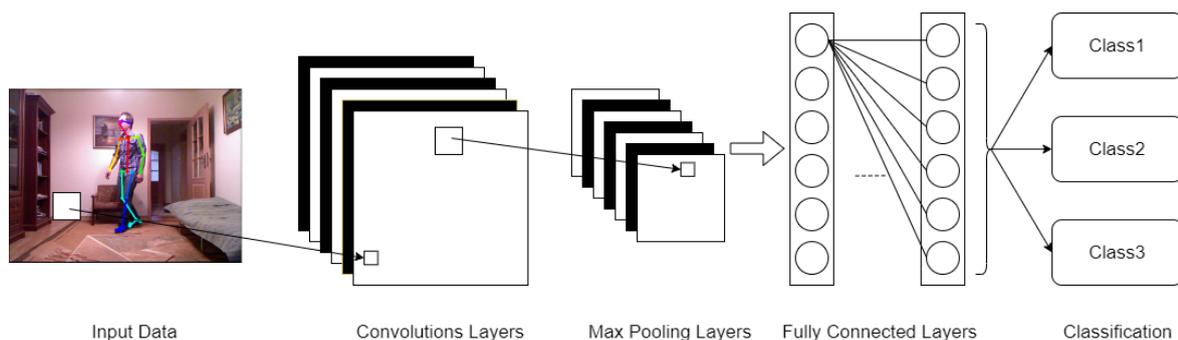


圖 2-2、卷積神經網路(CNN)[22]

卷積層是一組平行的特徵圖 (Feature Map)，它通過在輸入圖像執行一個元素對應乘積並求和的運算，對不同的卷積核執行運算而組成，產生所需的特徵，可以對微小的變化保持不變性 (invariance)，如旋轉、平移、伸縮等。如圖 2-3所示，一張 5×5 的圖片和一張 3×3 的卷積核 (Kernel) 做運算，在卷積層中移動的步數稱為 Stride 步數，Stride 為 1， \otimes 為卷積運算符號，之後產生出卷積特徵 (Convolved Feature)。黃色區塊為 3×3 圖片與卷積核進行運算，相同位置的元素互乘的總和為黃色區塊卷積特徵的新值。

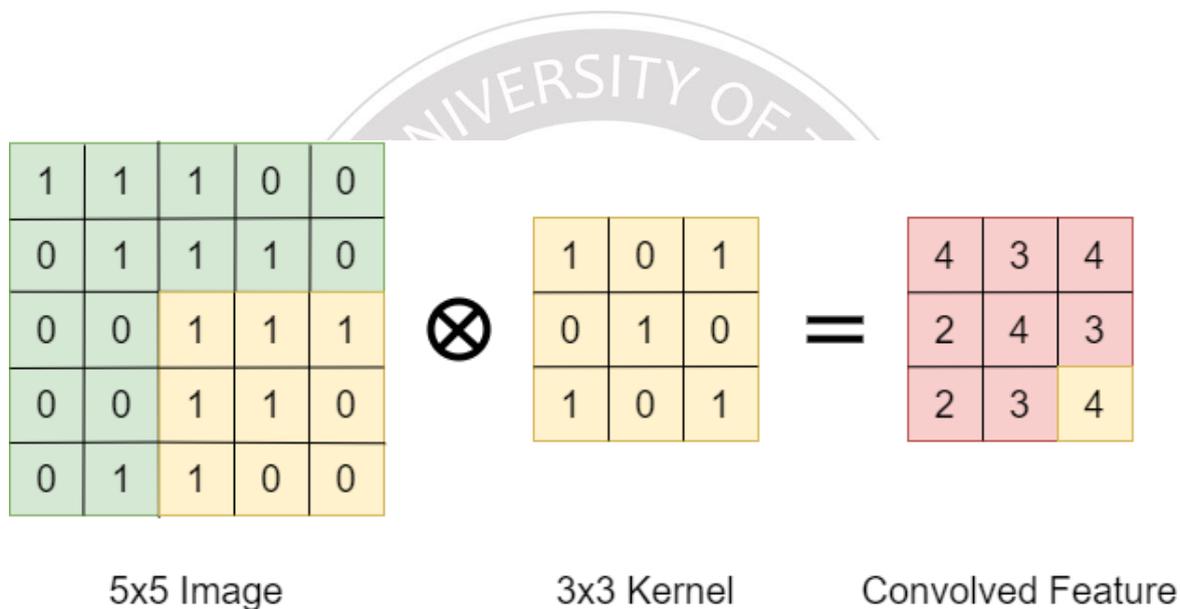


圖 2-3、卷積運算[22]

池化層跟卷積層相同，有卷積核(大多為 2×2)，對卷積層出來的特徵圖(Feature Map)做運算。目的是保留重要的特徵及降維，減少參數能防止過度凝合(Overfitting)。如圖 2-4所示，利用卷積核在特徵圖上，步數為2，進行最大池化(Max Pooling)的基本運算，在卷積核經過的地方，取出最大值，同時達成降維。

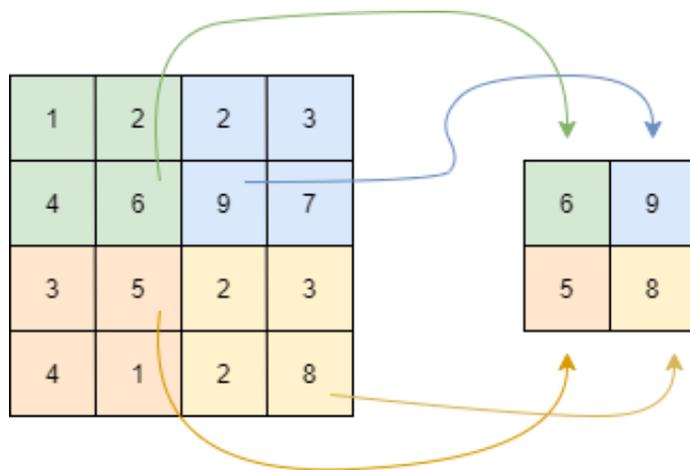


圖 2-4、最大池化運算[22]

2.1.3 遞歸神經網路(RNN)

遞歸神經網路(Recurrent Neural Network, RNN)[23-24]，也稱為循環神經網路，是神經網路的一種。遞歸神經網路能描述動態行為，如圖 2-5所示，和前饋神經網路(Feedforward Neural Network, FNN)接受較特定結構的輸入不同，RNN將狀態在自身網路中循環傳遞，因此可以接受更廣泛的時間序列結構輸入。

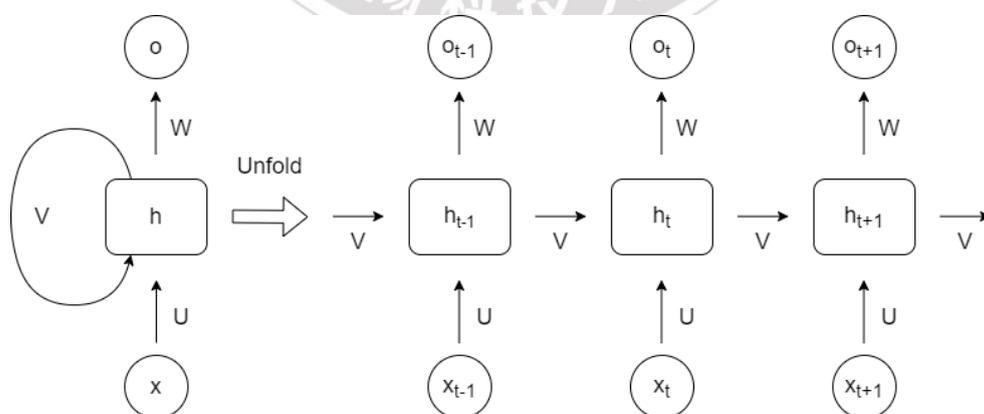


圖 2-5、遞歸神經網路(RNN)[25]

RNN與FNN之差異在於，RNN可以將 h_{t-1} 層計算完成的輸出，再回傳給 h_t 層作為輸入， o 為輸出向量，兩者分別定義為：

$$h_t = \sigma_h(Vh_{t-1} + U_h x_t + b_h) \quad (2.1)$$

及

$$o_t = \sigma_o(W_o h_t + b_o) \quad (2.2)$$

其中時間為 t ， h 為隱藏層的向量， x 為輸入向量， V, W 及 U 為參數權重， b 為偏權值， σ_h 和 σ_o 都為激勵函數。

2.1.4 長短期記憶 (LSTM)

長短期記憶(Long Short-Term Memory, LSTM)，是RNN的演化，由S. Hochreiter and J. Schmidhuber於(1997)首次提出[26]。時常被運用在語音辨識[27]、股價預測[28]和行人軌跡[29]等。LSTM改善了RNN長期記憶所發生的問題，增加了細胞狀態的概念，如圖 2-6所示，LSTM增加了四個結構，其中包含遺忘閘門、細胞狀態、輸入閘門及輸出閘門等，透過這些閘門將重要得特徵保留。

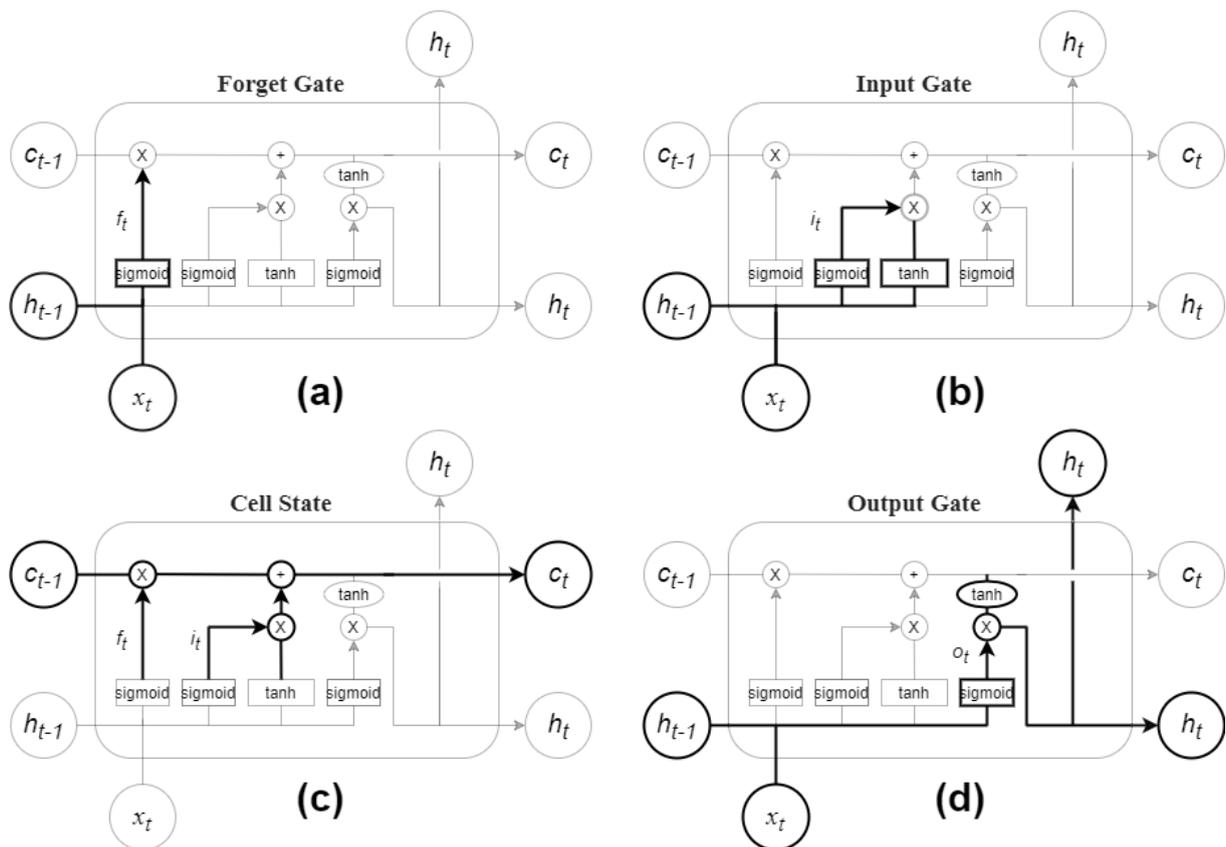


圖 2-6、LSTM[25]

LSTM的主要核心就是細胞狀態，通過設計被稱為閘門的結構，刪除或增加資訊到細胞的能力。遺忘閘門 f_t 能決定細胞狀態中需要丟棄的資訊，定義為：

$$f_t = \sigma_g(W_f x_t + U_f h_{t-1} + b_f) \quad (2.3)$$

其中小寫變量為向量，矩陣 W_e 為輸入權重，矩陣 U_e 為循環連接的權重， b_e 為偏權值，下標 e 可為 f, i, o, c 任一Gate。 t 為時間， h 為輸出向量， σ_g 為Sigmoid function。觀察Forget Gate's 激活向量 f_t ，可以判定是否保留該資訊。

知道有什麼新資訊將進入細胞的為輸入閘 i_t ，而要為新候選值激活向量為 \tilde{c}_t ，兩者分別定義為：

$$i_t = \sigma_g(W_i x_t + U_i h_{t-1} + b_i) \quad (2.4)$$

及

$$\tilde{c}_t = \sigma_h(W_c x_t + U_c h_{t-1} + b_c) \quad (2.5)$$

其中 σ_h 為tanh function。

細胞更新透過 c_t 來計算， c_t 為細胞狀態向量， o_t 為輸出閘門的激活向量， h_t 為當前輸出向量，三者定義為：

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t \quad (2.6)$$

，

$$o_t = \sigma_g(W_o x_t + U_o h_{t-1} + b_o) \quad (2.7)$$

及

$$h_t = o_t \sigma_h(c_t) \quad (2.8)$$

其中 \odot 為Hadamard乘積。

2.1.5 門控遞歸單元(GRU)

門控遞歸單元(Gated Recurrent Unit, GRU)[30][31]是遞歸神經網絡的門控機制，由Kyunghyun Cho等人於2014年引入[32]，為了改善LSTM執行速度較慢的問題，用來加快執行速度及減少記憶體耗用，同時在龐大的數據集中，有相當不錯的表現。如圖 2-7所示、GRU 將 LSTM 中的遺忘閥 (Forget Gate) 與輸入閥 (Input Gate) 用一個更新閥 (Update Gate) 取代，並把單元狀態 (Cell State) 和隱藏狀態 (h_t) 進行合併，因此其參數比LSTM少。

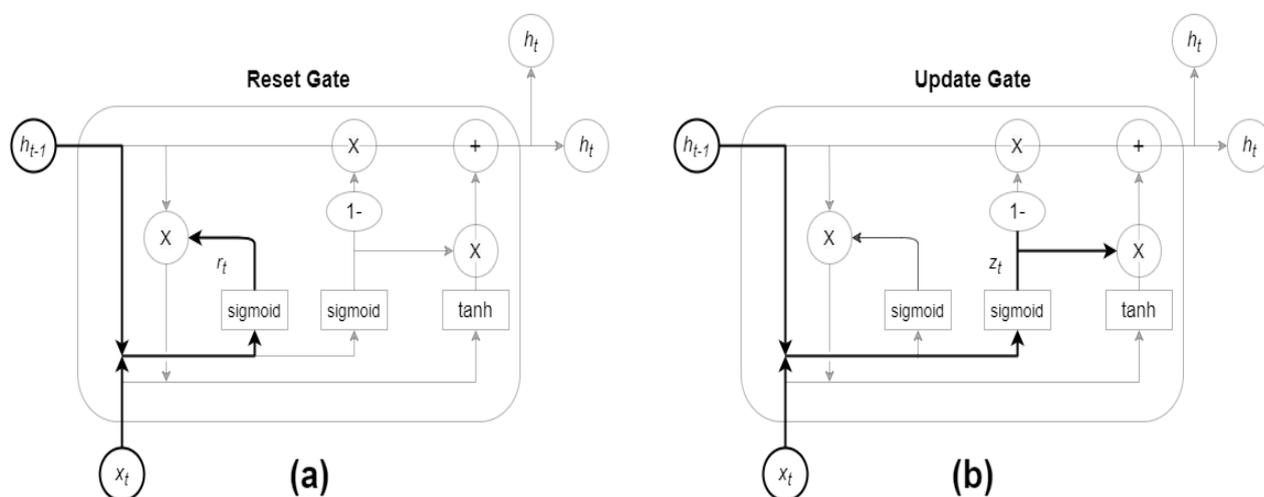


圖 2-7、GRU[25]

GRU 背後的原理與 LSTM 非常相似，使用 Gate 機制控制輸入、記憶等信息。GRU 有兩個 Gate，一個為重置門 (Reset Gate) 另一個為更新門 (Update Gate)。

重置閘門會決定新的輸入信息與時間 $t - 1$ 的記憶相結合。時間為 t ， z 為更新門的向量定義為：

$$z_t = \sigma_g(W_z x_t + U_z h_{t-1} + b_z) \quad (2.9)$$

其中 W 和 U 為權重參數， b 為偏權值， σ_g 為Sigmoid function。

更新門定義了時間 $(t - 1)$ 記憶保存到當前時間 t 的量， r 為重置閘門向量， h 為輸出向量，將保留的特徵值傳遞給下一時間，兩者定義為：

$$r_t = \sigma_g(W_r x_t + U_r h_{t-1} + b_r) \quad (2.10)$$

及

$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \sigma_h(W_h x_t + U_h (r_t h_{t-1}) + b_h) \quad (2.11)$$

其中 \odot 為Hadamard乘積， σ_h 為Tanh function。

2.2 激勵函數

激勵函數是神經網路中相當重要的一部份，它能引入非線性，當輸入及輸出無法脫離線性時，神經網路將無法達到有效的學習。

2.2.1 Logistic Sigmoid (Sigmoid)

Sigmoid是神經網路很常見的激勵函數，為S形圖形，能將輸入限制在0到1的範圍內輸出，可表示為

$$f(x) = \sigma_g(x) = \frac{1}{1+e^{-x}} \quad (2.12)$$

x 為輸入層輸入的數據。

2.2.2 Hyperbolic Tangent Sigmoid (Tanh)

Tanh能將輸出範圍限制在1到-1之間，可表示為

$$f(x) = \sigma_h(x) = \frac{(e^x - e^{-x})}{(e^x + e^{-x})} \quad (2.13)$$

x 為輸入層輸入的數據。

2.2.3 Rectified Linear Unit (Relu)

Relu是廣泛被使用的激勵函數之一，當 x 大於0時，輸出 x ，當 x 小於等於0時，輸出0，可表示為

$$f(x) = \begin{cases} 0 & \text{for } x \leq 0 \\ x & \text{for } x > 0 \end{cases} \quad (2.14)$$

x 為輸入層輸入的數據。

2.3 最佳化演算法

最佳化演算法是用來減少預測值與目標值之間的損失，用其找到損失值的最小值，讓神經網路提高準確率。

2.3.1 Gradient Descent

是一種一階導數的最佳化演算法，是相當經典的最佳化方法之一。如(2.15)式所示根據隨機的起始點 w ， t 為時間，透過不斷調整學習率 η 大小及梯度 ∇L 。往Loss的最小值去尋找得出更新模型的參數 w^{t+1} 。

$$\begin{aligned} w^{t+1} &= w^t + \eta(-\nabla L(w^t)) \\ &= w^t - \eta \nabla L(w^t) \end{aligned} \quad (2.15)$$

2.3.2 Adaptive Moment Estimation(Adam)

Adam[33]是經改良過的最佳化算法之一，計算梯度的一階矩估計 m_t ，二階矩估計 n_t ，兩者定義為：

$$m_t = um_{t-1} + (1 - u)g_t \quad (2.16)$$

及

$$n_t = vn_{t-1} + (1 - v)g_t^2 \quad (2.17)$$

其中 u 為一階矩估計指數衰減率， v 為二階矩估計指數衰減率， t 為時間， g 為梯度

透過上述公式能得到 m_t 和 n_t ，但需要校正因此要計算校正後的一階矩估計 \hat{m}_t 和二階矩估計 \hat{n}_t ，算出更新模型的參數 $\Delta\theta_t$ ，三者定義為：

$$\hat{m}_t = \frac{m_t}{1 - u^t} \quad (2.18)$$

，

$$\hat{n}_t = \frac{n_t}{1 - v^t} \quad (2.19)$$

及

$$\Delta\theta_t = -\frac{\hat{m}_t}{\sqrt{\hat{n}_t + \epsilon}}\eta \quad (2.20)$$

其中 η 為學習率。

2.4 損失函數

Loss function 也稱為Object function，是用來計算神經網路的預測值和目標值之間的差距，Loss function越小代表神經網路準確度越好。

2.4.1 Mean Square Error(MSE)

MSE常被運用在回歸問題上定義為：


$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad (2.21)$$

其中 n 為資料數量， y_i 為目標值， \hat{y}_i 為預測值。

2.4.2 Cross Entropy

Cross Entropy常被運用在分類問題。 p 為真實事件， q 為事件發生機率
定義為：

$$H = \sum_{x=1}^X \sum_{i=1}^N -p(x_i) \log_2 q(x_i) \quad (2.22)$$

其中 N 為測試集大小， X 為分類類別。

2.5 模型評估指標

人工神經網路、機器學習或是專家系統等領域，評估是一項重要的工作，常見得評估指標[34]如：準確率(Accuracy)、精確率(Precision)、召回率(Recall)和F1-Measure。如圖 2-8所示，透過混淆矩陣(Confusion Matrix)來計算評估指標，檢測模型分類效果。

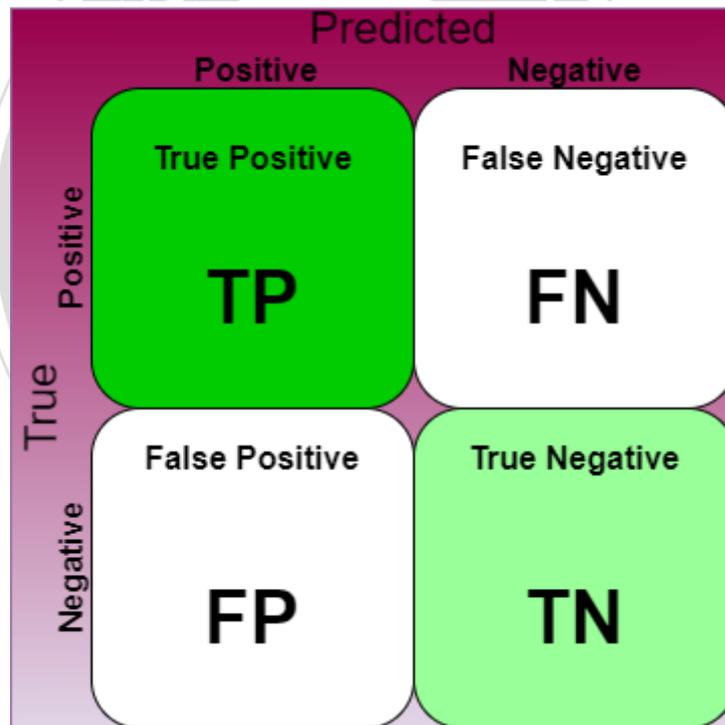


圖 2-8、混淆矩陣[34]

圖 2-8 中的 Positive 為主要分類的目標，Negative 為次要分類的目標，而 True 和 False 分別為結果預測正確及錯誤。分為四種預測情況：(True Positive, TP)、(True Negative, TN)、(False Positive, FP) 和 (False Negative, FN)。TP 為正確預測主要目標，TN 為正確預測次要目標，FP 為錯誤預測主要目標，FN 為錯誤預測次要目標。

2.5.1 靈敏度(Sensitivity)

是指實際為主要分類目標(陽性)的所有樣本中，預測為主要分類目標的比例。如真實事件為跌倒(陽性)的所有事件中，模型預測也為跌倒，兩者的比例：

$$S = \frac{TP}{TP+FN} \times 100\% \quad (2.23)$$

2.5.2 特異度(Specificity)

是指實際為次要分類目標(陰性)的所有樣本中，判斷為次要分類目標的比例。如真實事件為非跌倒(陰性)的所有事件中，模型預測也為非跌倒，兩者的比例：

$$F = \frac{TN}{TN+FP} \times 100\% \quad (2.24)$$

2.5.3 準確度(Accuracy)

是指所有分類的樣本中，判斷正確為主要和次要分類目標的比例。如所有事件，模型正確預測跌倒和非跌倒的比例。

$$A = \frac{TN+TP}{TP+TN+FP+FN} \times 100\% \quad (2.25)$$

2.6 OpenPose提取骨架

OpenPose即時多人2D姿勢估計，是由美國卡內基梅隆大學(CMU)研發，是一種基於監督式卷積神經網路，以Caffe為框架開發的。可以實現人體動作、面部表情、手指運動等姿勢估計。適用於單人和多人，具有極好的辨識效果，並且辨識速度相當快速。

OpenPose演算法是通過部分親和力區域(Part Affinity Fields, PAFs)和置信圖(Confidence Map)計算出人體骨架。首先，先經過VGG-19模型得到特徵圖(Feature Map)，之後使用兩分支多階段的CNN網路，把先前計算的特徵圖輸入。第一分支為預測關節部位的一組2D置信圖，第二分支為PAFs，是預測肢體部位，最後，通過貪婪演算法解析置信圖和PAFs，對圖中的人物生成2D骨架。

但是，OpenPose依然是使用2D圖像的特徵來取得骨架，所以當缺少這些特徵時，會產生無法預測或是預測錯誤等問題。像是人體重疊、背景導致人體邊緣不明顯或是頭部低於臀部時，都會發生骨架無法估計或錯誤。

2.7 正規化(Normalization)

是將原始的數據照比例縮放於設定的區間中，且不改變原本的分布。做正規化的好處是能提升模型的收斂速度，並提高模型的準確度。每個數據集都有適合正規化方法。

2.7.1 Z分數標準化(Z-Score Standardization)

經Z分數標準化後，資料將符合標準常態分佈(Standard Normal Distribution)，轉換後的平均值=0、標準差=1，且用標準分數或稱Z分數(Z-Score)來作為單位定義為：

$$Z = \frac{X - \mu}{\sigma} \sim N(0,1) \quad (2.26)$$

其中 X 為原始數據， μ 為平均數， σ 為標準差。

2.7.2 最小值最大值正規化(Min-Max Normalization)

最小值最大值正規化的用法，是將資料等比例縮放到 $[0, 1]$ 區間中，避免數值過大，造成梯度爆炸，新比例的資料為 X_{nom} 定義為：

$$X_{nom} = \frac{X - X_{min}}{X_{max} - X_{min}} \in [0,1] \quad (2.27)$$

其中 X 為原始數據， X_{min} 為數據中的最小值， X_{max} 為數據中最大值。

2.8 插值方法

在數值分析的領域中，內插法或稱插值(Interpolation)是一種通過已知的、離散的數據點，在範圍內推算新數據點的過程或方法。而根據這些數

據，我們希望得到一個連續的函數(也就是曲線)。

2.8.1 線性插值

線性插值採用兩個數據點，計算兩點彼此的斜率，來推算出兩點之間的值， Y 為推測值定義為：

$$Y = Y_0 + \frac{Y_1 - Y_0}{X_1 - X_0} (X - X_0) \quad (2.28)$$

其中 X_0 、 Y_0 、 X_1 和 Y_1 為數據集中的已知數據， X 為 X_0 和 X_1 之間的任何數據。

線性插值是快速簡單的，但精確度不高。

2.9 文獻探討

本節是依據本研究所需之領域進行探討，將探討跌倒檢測相關文獻，以了解其特性及應用。

2.9.1 穿戴式裝置跌倒檢測

本小節將探討穿戴式裝置跌倒檢測相關領域文獻[6-7]、[40-42]，主要是使用陀螺儀傳感器(Gyro Sensors)、心律傳感器(Fusion of Heart Rate Sensor)、可穿戴生物醫學信號測量終端(Wearable Biomedical Signal Measurement Terminal)或是加速度計(Accelerometer)來當作特徵值和判斷值。使用系統進行判斷或是結合機器學習(Machine Learning)如：決策樹(Decision Tree, DT)、最近鄰居法(K-Nearest Neighbor, KNN)和支援向量機(Support Vector Machine, SVM)來檢測跌倒，這種跌倒檢測方式準確度較高，因為傳感器的數值準確度高並且即時性佳，因此在後續的判斷或分類能有不錯的效果，針對穿戴者的隱私權問題又有保障。

其缺點是需穿戴感測器，有穿戴者遺忘穿戴、裝置損壞及適應不良

等問題出現，都會造成無法檢測的情況。另一狀況為當發生跌倒時，裝置的大小可能會造成穿戴者有更嚴重的傷害，因此需要一種可以不穿戴感測器的跌倒檢測方法。

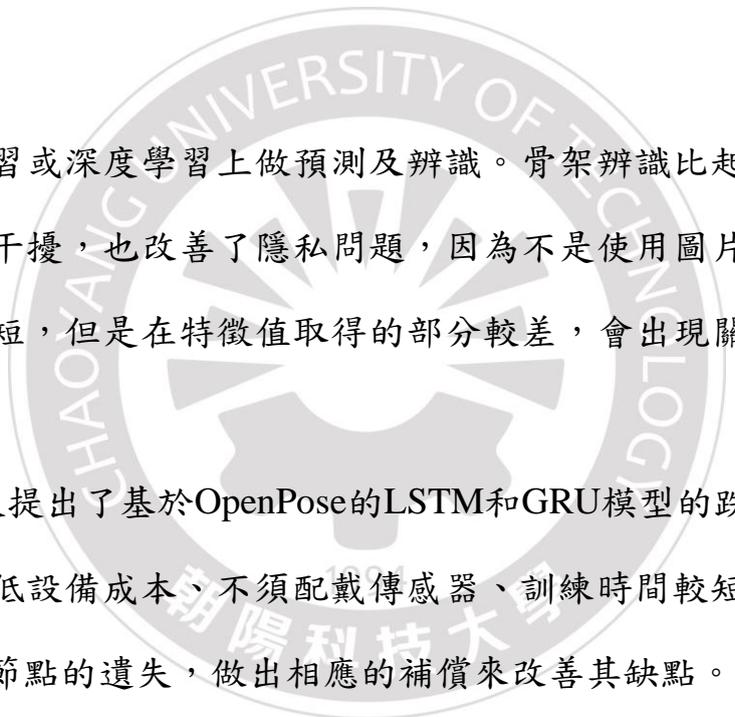
2.9.2 環境感測裝置跌倒檢測

此小節將探討環境裝置跌倒檢測相關領域文獻[8]、[43-47]，主要是使用像地板震動、聲音或是紅外線感測裝置來當作特徵值和判斷條件，使用系統進行判斷，或結合機器學習及類神經網路，此跌倒檢測方法比起穿戴式裝置跌倒檢測方法準確度來的低一些，因為感測範圍較大，容易受到環境干擾所影響，同時也需要較多的設備成本來建置，隨著房屋的大小和擺設而有所不同。

2.9.3 電腦視覺跌倒檢測

此小節將探討電腦視覺跌倒檢測相關領域文獻，首先是透過人體輪廓的跌倒檢測方法[10]、[35]、[48-50]，為了分離運動和局部對象的運動，提取僅具有運動對象的前景圖像，使用高斯混合模型(Gaussian Mixture Model, GMM)創建背景模型，對前景圖二值化後，二值化可得出當前的人體輪廓。在執行背景扣除之後，將獲得人體輪廓，使用機器學習或類神經網路來進行分類，建置成本低，也不會有感測器穿戴的問題，但是此方法容易受到環境干擾，如：光線、服裝特徵或是人像重疊，同時也會造成隱私權的暴露。

第二、為透過人體骨架的跌倒檢測方法[9]、[51-52]，使用Kinect、OpenPose或是Deepcut[36]等方式來找出人體骨架，取得到的關節點位，將

The logo of Shaoyang University of Technology is a circular emblem. It features a central gear-like shape with a sunburst or starburst pattern in the center. The words "SHAOSYANG UNIVERSITY OF TECHNOLOGY" are written around the perimeter of the circle. Below the circle, the Chinese characters "韶陽科技大學" are visible.

被用在機器學習或深度學習上做預測及辨識。骨架辨識比起人體輪廓辨識較不易被環境干擾，也改善了隱私問題，因為不是使用圖片進行訓練，因此訓練時間較短，但是在特徵值取得的部分較差，會出現關節點遺失的問題。

因此，本文提出了基於OpenPose的LSTM和GRU模型的跌倒檢測方法，此方法可以降低設備成本、不須配戴傳感器、訓練時間較短和保護隱私權，本文也針對關節點的遺失，做出相應的補償來改善其缺點。

第三章 基於OpenPose骨架的LSTM/GRU模型

本章將針對提出的正規化方法和跌倒檢測模型做說明，並介紹模型的預處理及訓練流程。首先3.1節為模型訓練流程之說明；3.2節為數據集說明；在3.3節的部分說明人體骨架取得方式；3.4節為預先處理之說明；3.5節為神經網路架構之說明；3.6節為實驗環境之說明。

3.1 模型訓練流程

如圖 3-1所示，模型的訓練分為四個部分，骨架取得、預先處理、模型訓練及測試。

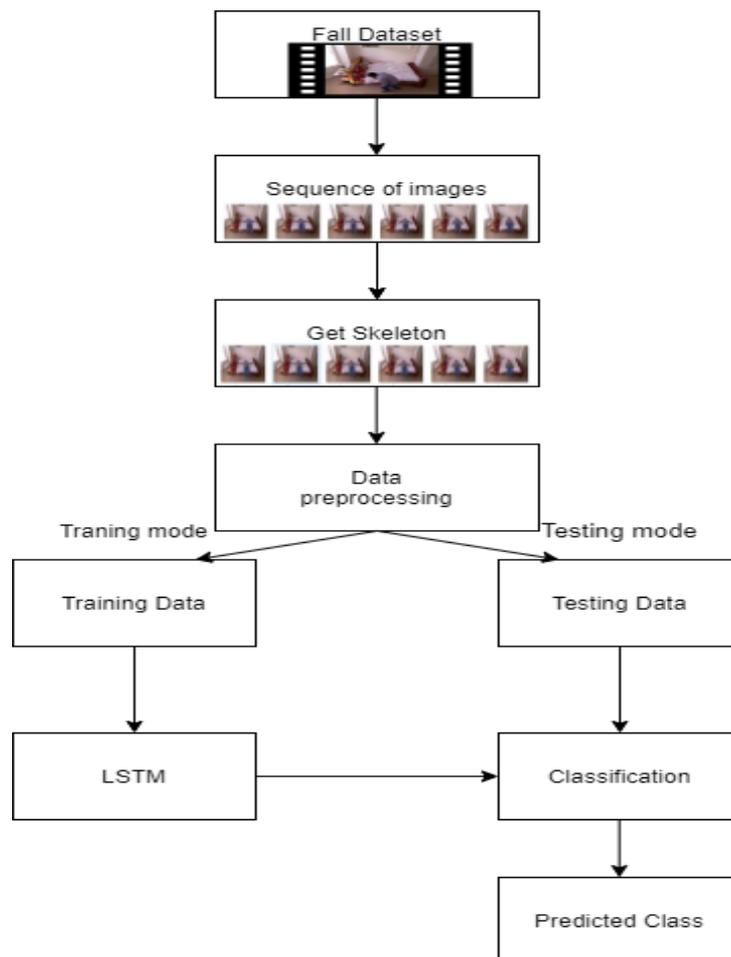


圖 3-1、模型訓練流程

第一部份將數據集的影片分割為連續圖像序列。第二部份將圖片序列使用OpenPose取出骨架，將得到25個人體關節點(x,y)。第三部份數據的預處理，如：最小最大值正規化或是我們提出的正規化方法。最後的部分是將數據集分為測試集和訓練集，輸入到模型中訓練，完成模型訓練後，就能進行分類的測試。

3.2 數據集

3.2.1 UR Fall Detection Dataset

我們使用UR Fall Detection Dataset[37]，它來自熱舒夫大學計算建模跨學科中心所製作，內容包含70個(30個跌倒和40個日常生活活動)序列，每秒30張圖進行拍攝。在跌倒事件中，記錄了發生到倒下的情況，而日常生活則有許多情況，像是站立、蹲下撿物品和躺下。環境有燈光充足和偏暗等情況，他們使用2台Microsoft Kinect相機，與相應的加速度計數據記錄跌倒事件，跌倒事件使用兩台設備，日常活動事件僅用一台設備(相機0)和加速度計記錄。如圖 3-2所示，設備位置高度大約在人體腰部的高度，影像尺寸為640×480，我們將這些圖像序列以100張為一組，做了資料增強後，共有792組。

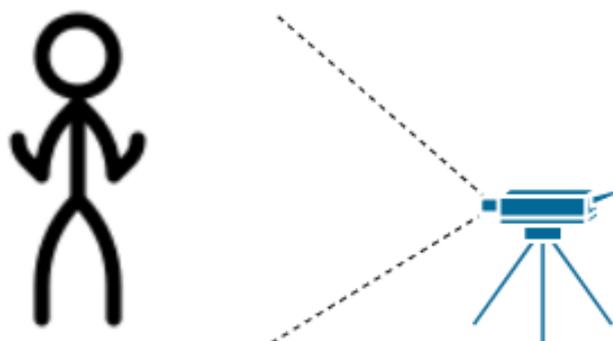


圖 3-2、UR Fall Detection Dataset設備位置

3.2.2 Fall Detection Dataset

在Fall detection Dataset[38]中，記錄在由8個不同視角組成的5個不同房間中。有5位不同的參與者，其中有兩名男性參與者和三名女性參與者。參與者的所有活動都代表5種不同的姿勢，這些姿勢包括站立，坐著，躺著，彎曲和爬行，以每秒30張圖進行拍攝。如圖 3-3所示，設備位置高度大約在人體視線仰角45度。原始影像尺寸為320*240，調整影像尺寸為640*480，圖像序列以100張為一組，做了資料增強後，共有936組。

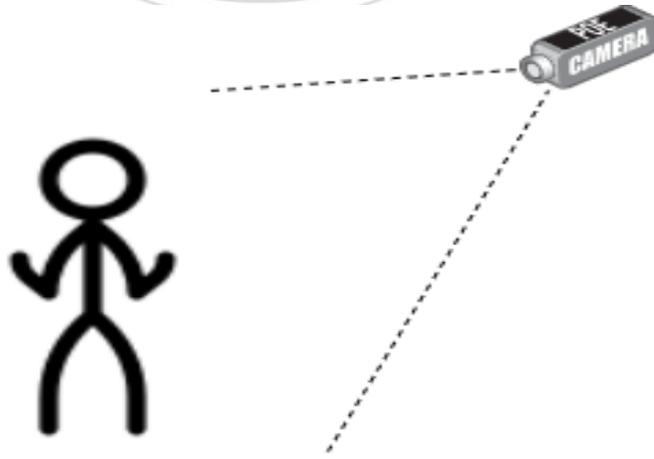


圖 3-3、Fall Detection Dataset設備位置

我們混合了這兩個數據集，一方面是確保有一定的數據量，另一方面是為了避免過擬合(Overfitting)，使用不同視角的數據集，增加樣本多樣性。如表 3-1所示，我們個別從數據中取出部份數據，為了保持公平性，跌倒事件和非跌倒事件的數量相同，本文將使用Mix Dataset來當作模型訓練的數據集。

表 3-1、數據集分類數量

數據集	事件類型	非跌倒事件 (組)	跌倒事件 (組)
UR Fall Detection Dataset		240	240
Fall Detection Dataset		330	330
混合數據集		570	570

3.3 OpenPose取得人體骨架

本文使用OpenPose演算法來抓取骨架，如圖 3-4所示，為OpenPose估計出的人體關節點，共25個關節點。0為鼻子，(15~18)為右左眼睛及耳朵，1為頸部，(2~7)為右左肩膀、手肘和手腕，8為臀圍中央，(9~14)為右左臀圍、膝蓋和腳踝，(19~24)為左右腳掌、腳趾及腳跟。

如圖 3-5所示，為OpenPose抓取的關節點，能發現此方法能在複雜的空間環境中，如：昏暗的環境，也能找出人體關節點。一張圖片會得一個 $M_{3 \times 25}$ 的矩陣，此矩陣為25個關節點的 (x, y) 座標和關節點位的得分，若得分越高表示關節點位置準確度越高。

3.4 數據預先處理

數據的預先處理是相當重要的部份，因為最初取得的數據未必有相同格式或單位，因此需要調整格式、去除異常值、填補缺值或對特徵進行縮放。為了讓模型有更好的學習成效。此章節將分為三個部份，第一部份調整格

式，第二部份正規化，第三部份填補缺值。

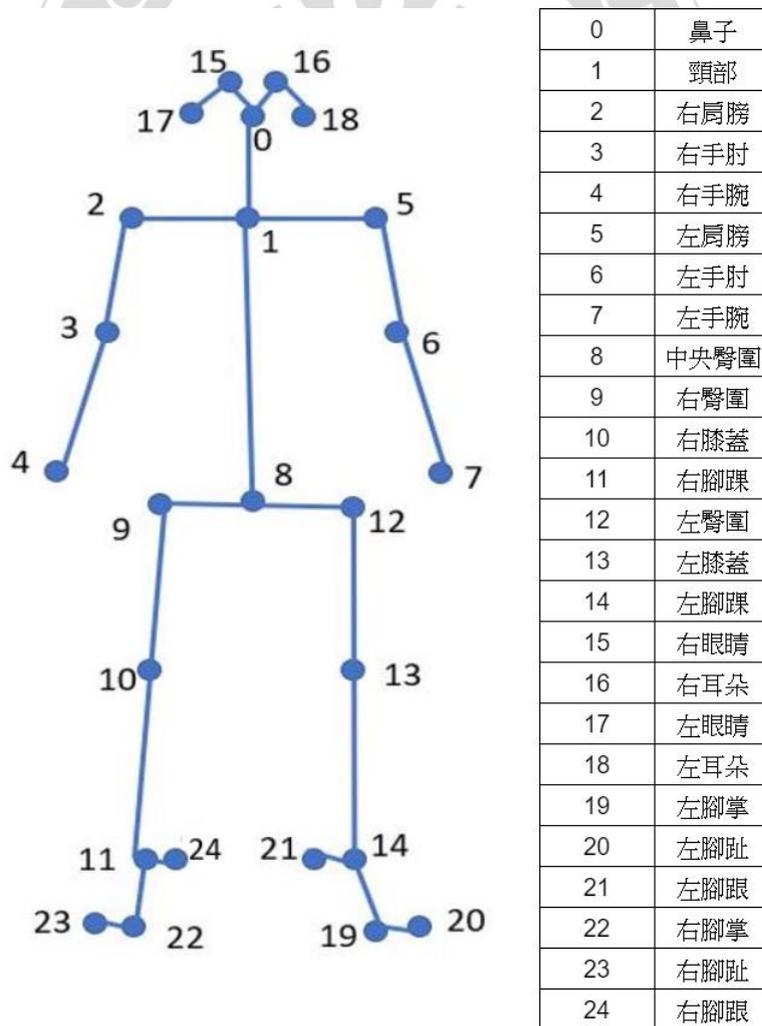


圖 3-4、人體骨架關節點(25點)

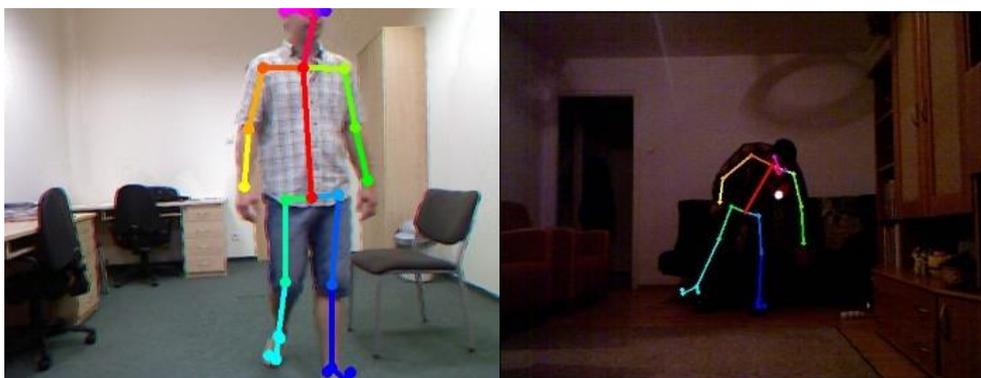


圖 3-5、OpenPose抓取到的關節點位

3.4.1 格式調整

將原本得到的 $M_{3 \times 25}$ 的矩陣，去除掉關節點位得分，將變成 $M_{2 \times 25}$ 的矩陣，因為以100張圖為1組，共1140組，但遞歸神經網路的基本輸入維度是3維，所以必須將 $M_{2 \times 25}$ 矩陣轉為 $M_{1 \times 50}$ 的一維矩陣，才可以輸入。

3.4.2 相對位置正規化

為了提升模型準確率，本文嘗試使用最大最小值正規化，來提升模型收斂速度，但數據集中有遺失點存在，其值為0表示，所以會發生原始數據分佈被改變，造成特徵不明確和複雜，因此我們提出一種將原始座標位置，移動到我們設相對座標位置的正規化方法。人體骨架關節點為 (x_n^f, y_n^f) ， $n = \{0, 1, 2, \dots, 14\}$ 為關節點編號， $f = \{0, 1, 2, \dots, 99\}$ 為圖片編號，將被位移到本文設置的中心點 (x_c^f, y_c^f) 周圍定義為：

$$x_c^f = \frac{W}{2} \quad (3.1)$$

及

$$y_c^f = \frac{H}{2} \quad (3.2)$$

其中當 x_n^f 或 y_n^f 等於0時，將它視為遺失點不計算，若中心點為遺失點，將由計算出的中心點直接取代， W 為圖片寬度， H 為圖片高度。

必須先計算需要位移的方向和距離 (x_{dis}, y_{dis}) 定義為：

$$x_{dis}^f = x_8^f - x_c^f \quad (3.3)$$

及

$$y_{dis}^f = y_8^f - y_c^f \quad (3.4)$$

其中第8關節點為 (x_8^f, y_8^f) 中央臀圍做為基準點，位移到中心點 (x_c^f, y_c^f) ，當位移值小於0表示向右，位移值大於0表示向左移動。

計算位移到相對位置的新關節點座標 (rx_n^f, ry_n^f) 定義為：

$$rx_n^f = (x_n^f - x_{dis}^f) \quad (3.5)$$

及

$$ry_n^f = (y_n^f - y_{dis}^f) \quad (3.6)$$

其中原始關節點 (x_n^f, y_n^f) ，需位移距離為 (x_{dis}^f, y_{dis}^f) 。

這麼做的好處有兩點，第一是避免了將遺失點一併計算，不會讓原始數據分佈被影響，第二是減少不必要特徵，因本文主題為跌倒檢測，所以去除了原始數據中，人體位移過程這個特徵，像是左走到右或是從遠方走到近，將人體骨架固定在相同位置，做出連續動作讓模型更容易學習。

如圖 3-6所示，我們將圖像大小統一為640*480並設置一個中心點，經過上述公式位移後，如圖 3-7所示，人體骨架將移動到圖像中央，中央臀圍關節點會與中心點位置相等。在跌倒及非跌倒活動骨架圖中，我們發現有些關節點位對跌倒事件的識別是不太重要的，所以將它去除，因過多的特徵參數也可能導致辨識率下降或過於複雜。只保留了15個重要點位當作訓練的特徵值，其中保留了鼻子、肩膀、手肘、手腕、頸部、臀圍、膝蓋及腳踝的關節點。

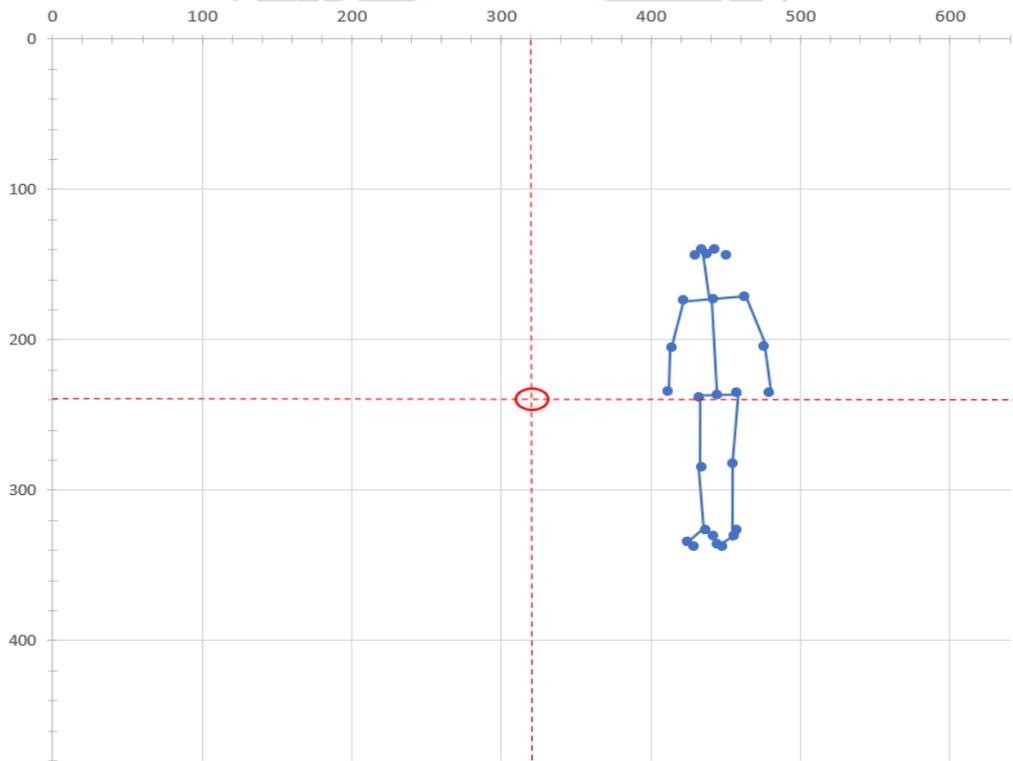


圖 3-6、原始的人體骨架座標位置及數量

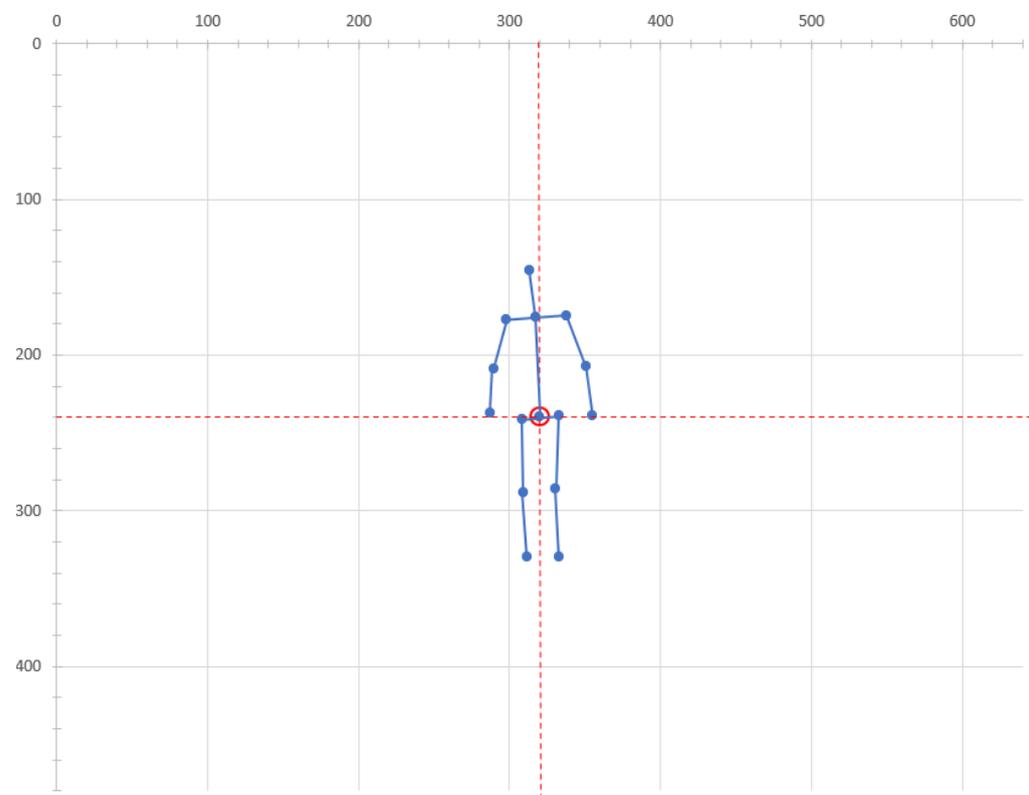


圖 3-7、位移後的關節點位置及數量

3.4.3 補償遺失值

如圖 3-8所示，當人體有出現重疊、被遮蔽、人體身形輪廓不明確或頭下腳上的情況出現，會造成在抓取骨架時，有遺失和錯誤發生。因此，本文使用線性的插值方法，來補償遺失值。我們設置100張圖為一組，利用原始的非遺失值數據來推算，因此要確保有足夠的數據。



圖 3-8、關節點遺失

本文使用了一種插值方法嘗試補值，此方法為線性插值。插值方法能透過原始數據來推估之後的數據值，若是該關節點的原始點為太少，將可能造成計算出的推測值誤差變大，因此本文設置一道門檻，若原始關節點 (x_n^f, y_n^f) 遺失超過67%，將不進行補值。

如圖 3- 9所示，本文以連續30張圖的手部關節點y點為例，做一個舉手再放下的動作，前15張為逐漸舉起手，從第16張圖手開始放下，嘗試計算補償值與原始值的誤差。

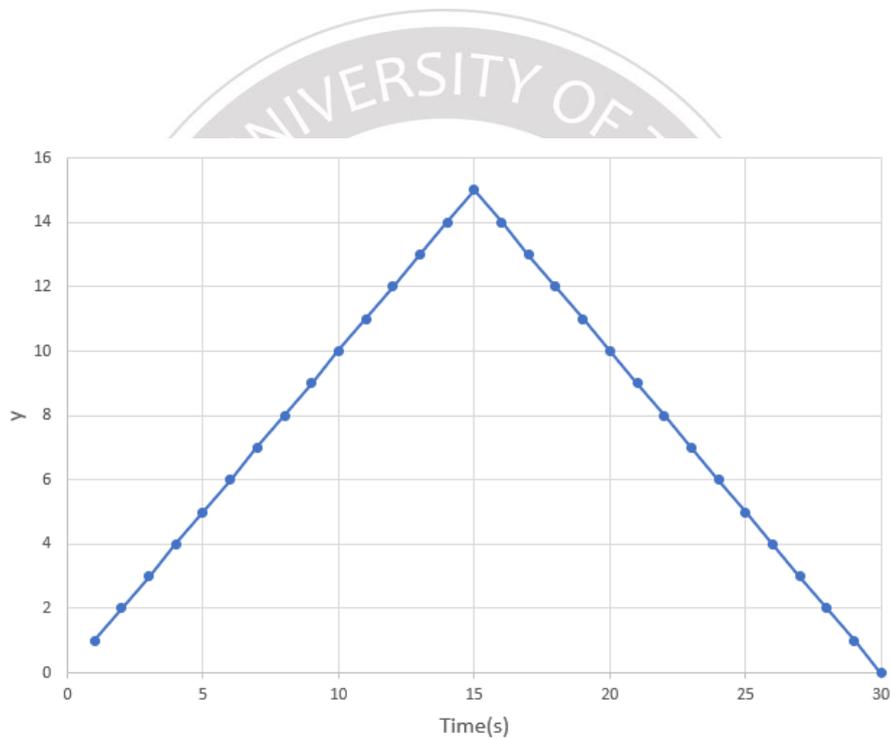


圖 3-9、手部y點變化

如圖 3-10和圖 3-11所示，遺失點補償並沒有想像中順利，因為有改變方向的節點存在，我們稱為轉折點，手向上舉起為一個慣性動作，因此能預測其方向，但是它何時會到最高點，或在最高點停留多久是很難推估的。看到圖中的紅色圈的部分為該動作的轉折點，由此點開始轉變方向，圖 3-10、線性插值補償遺失點為轉折點遺失，但圖序列14及16的y點沒有遺失，進行補值後因為與轉折點原始距離較近，因此誤差小於等於1。

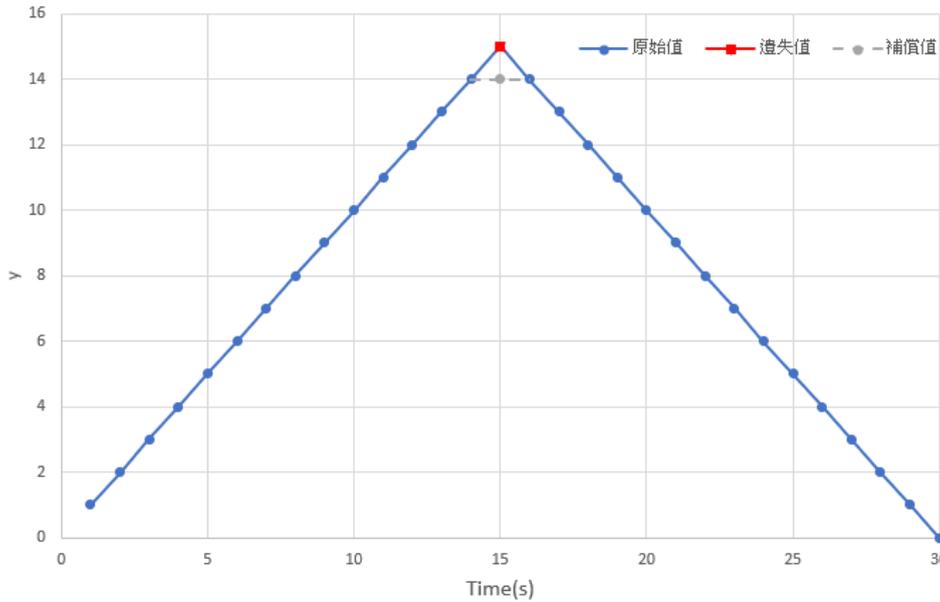


圖 3-10、線性插值補償遺失點(第15點遺失)

如圖 3-11所示，圖序列10到20的y點都遺失也包括轉折點，使用圖序列9和21點計算補償值，能看到誤差相當大，因為不知道轉折點確切位置導致插值的計算遇到很大的難題，最大誤差為5。而轉折點遺失的情況下，最大誤差為遺失轉折點減去補償轉折點。

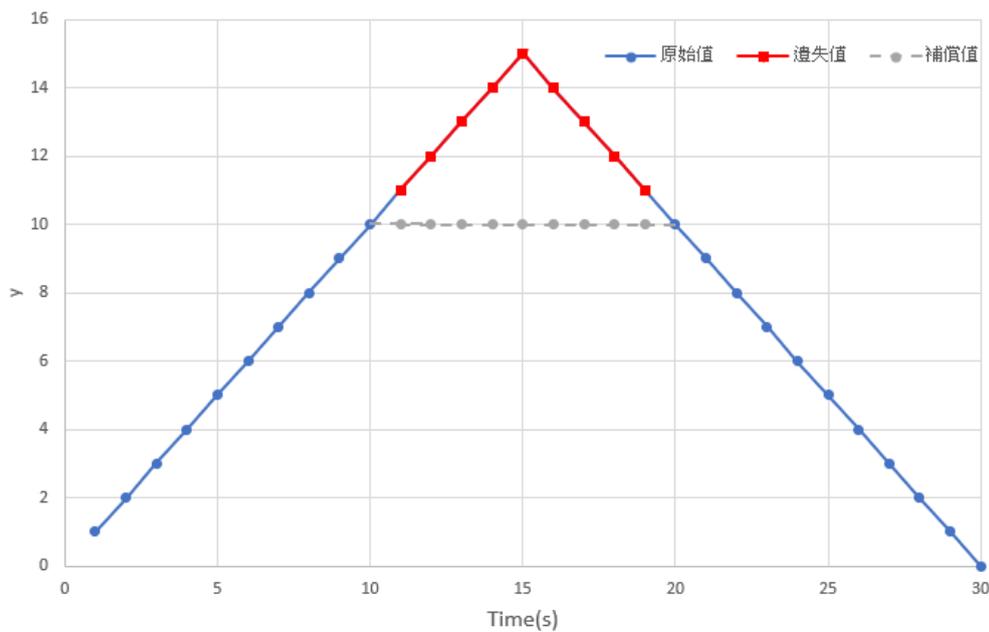


圖 3-11、線性插值補償遺失點(第11至19點遺失)

如圖 3-12所示，當轉折點不是遺失點時，將不會出現過大的誤差，估計值和真實值的誤差小於0。

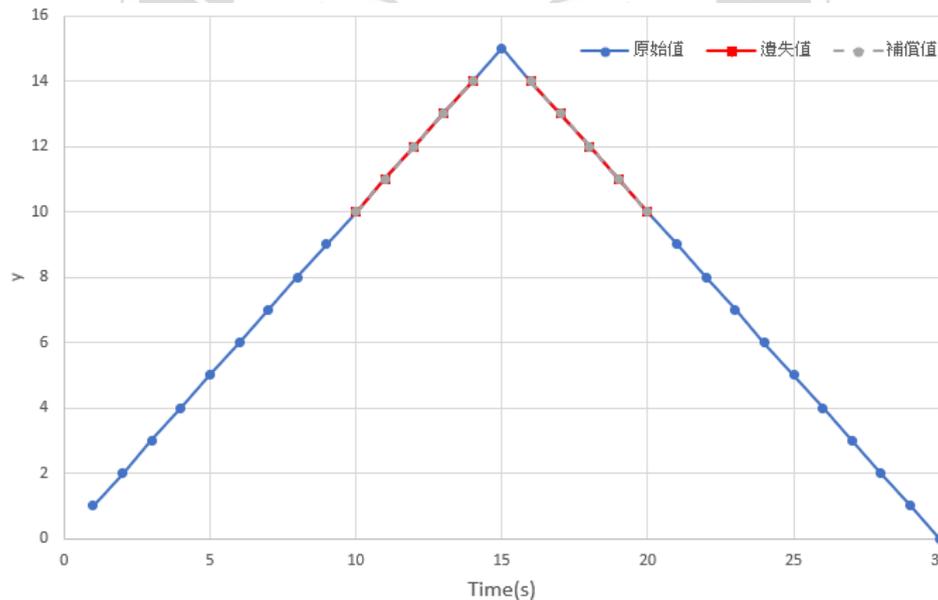


圖 3-12、線性插值補償遺失點(第10至14點, 第16至20點遺失)

如表 3-2所示，原始數據集共有2,850,000個關節點，而遺失點為520,112個，做了正規化後的數據集剩下15個關節，原始關節點有個1,710,000個，遺失點為214,707個。經過缺失值補償的關節點數量和正規化後的相同，而遺失點減少至為27,849個。其中，原始數據集的遺失關節點占了18.2%，做個正規後，去除了10個不重要的關節點，遺失關節點佔了12.5%，經過缺失值補償後，遺失關節點佔了1.6%。遺失點的減少能幫助模型在訓練時，減少噪音並增加準確度。

表 3-2、關節點遺失比例

數據集 \ 關節點	存在關節點	遺失關節點	占比
原始數據集	2,850,000	520,112	18.2%
相對位置正規化	1,710,000	214,707	12.5%
線性插值+相對位置	1,710,000	27,849	1.6%

3.5 模型架構

我們選用RNN、LSTM和GRU模型，來配合有時間特徵的關節點進行學習，因為遞歸神經網路相當適合處理與時間序列高度相關的問題，同時不必像使用純圖片偵測跌倒的方式一樣，計算整張圖片增加訓練時間及硬體負擔。而且人體關節的相連性，所以姿勢在時間 t 的變化上會有一定的規律。

如圖 3-13所示，我們設置了50個輸入，分別為25個關節點的 (x,y) 座標，作為特徵值輸入。設置1層隱藏層分別是RNN、LSTM和GRU，共三個模型。本文目標是分類問題，所以加入了Softmax層，它將多個神經元的輸出，歸一化到 $(0, 1)$ 區間內，如：最終是兩分類將會有兩個得分 $[0.7 \ 0.3]$ ，Softmax層會輸出最高得分的值，完成分類。

如圖 3-14所示，為設置了30個輸入，分別為15個關節點的 (x,y) 座標，作為特徵值輸入。設置1層隱藏層分別是RNN、LSTM和GRU，共三個模型，由於遞歸神經網路比起其他神經網路像是ANN結構較為複雜，因此不需要過多隱藏層，否則會讓模型過於複雜。加入了Softmax層，它將多個神經元的輸出，歸一化到 $(0, 1)$ 區間內，如：最終是兩分類將會有兩個得分 $[0.7 \ 0.3]$ ，Softmax層會輸出最高得分的值，完成分類。

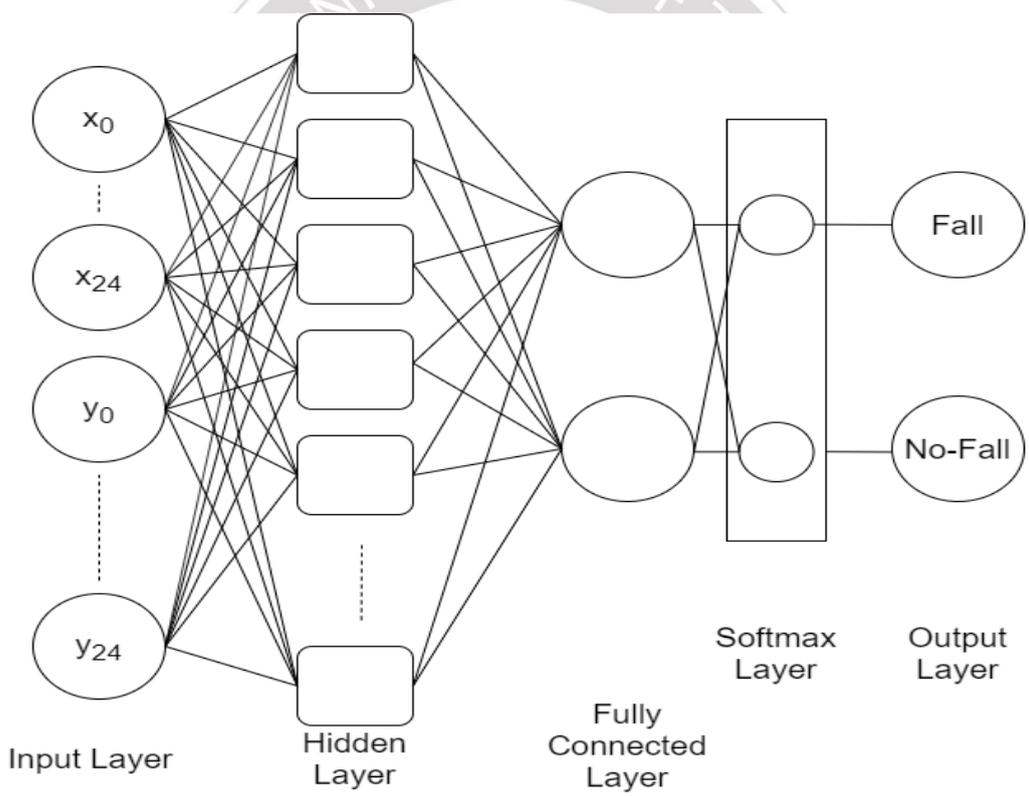


圖 3-13、跌倒模型架構(輸入50)

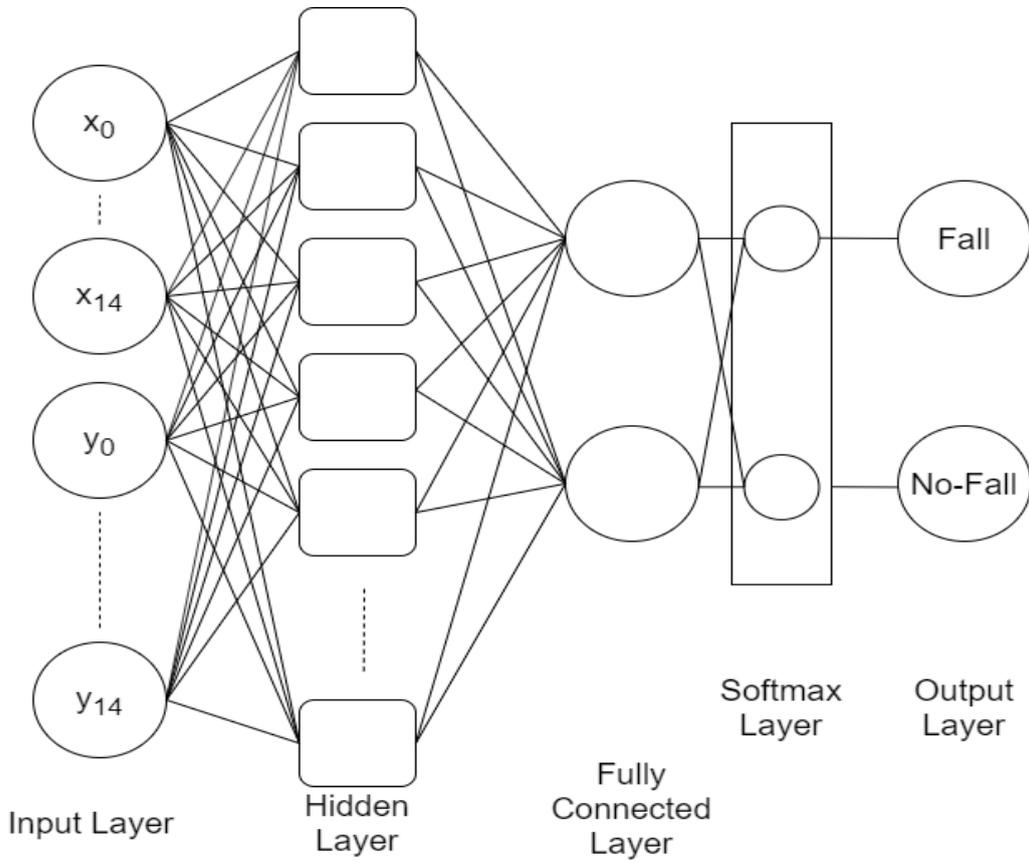


圖 3-14、跌倒模型架構(輸入30)

3.6 實驗環境

3.6.1 資料處理環境

本文數據集的處理與模型訓練是在不同環境下執行。預處理的部分包含骨架抓取、關節點補償及正規化等，使用硬體AMD R5 2600 CPU，記憶體為16GB，軟體Python3.6。

3.6.2 訓練環境

而神經網路模型訓練為Google Colaboratory平台[39]，使用CPU和GPU來做訓練，軟體環境為Keras 2.3.1。如表 3-3所示，本文使用三種不同的模型進行實驗，這三種模型分別為RNN、LSTM及GRU，三者都是遞歸神經網路，都擁有一個別擅長的處理能力，因此本文研究哪個模型，能有最有效檢測跌倒的效果。

在訓練參數的部分，本文將1140組的混合數據集隨機分為80%訓練集和20%測試集，會得到916個訓練集跟224個測試集，訓練集會經由批次的設置，而投入每次設置數量的訓練資料進去學習，本文設置為916與訓練集大小相同，原因是能讓模型更快的找到通往最佳值的梯度，但若資料過多或是硬體設備較低階，可能發生記憶體不足等問題。第二是學習率的部分，在不同學習率下的結果，學習率為尋找最佳值的步伐大小，若值太大將會找不到最佳值出現震盪，若太小訓練速度將會下降，因此需要找到最合適的學率。

表 3-3、模型訓練參數

模型 \ 訓練參數	學習率	批次大小	Epochs
RNN	0.1, 0.01, 0.001	916	500
LSTM			
GRU			

如表 3-4，有四種不同處理的數據集，分別為原始數據集、最小最大值正規化、相對位置正規化和線性插值加上相對位置。根據不同數據集，我們設置了50和30個輸入，分別為原始的25個關節點及15個關節點的(x,y)座標，作為特徵值輸入。設置隱藏層內分別有64至1024個隱藏神經元，因為不同數據集所需的隱藏層，及隱藏神經元得數量都不同，因此需要嘗試某範圍的數量，來找出最佳選擇，本為使用Tanh來當作模型的激勵函數，因有遺失點的存在，所以需要使用函數輸出是以0為中心的激勵函數，這樣輸入為0時，輸出也是0。

表 3-4、模型結構參數

數據集 \ 結構參數	未正規化	最大最小值 正規化	相對位置 正規化	線性插值+ 相對位置
輸入層 神經元數量	50		30	
隱藏層 神經元數量	64, 128, 256, 512, 1024			
激勵函數	Tanh			
最佳化算法	Adam			
損失函數	Cross Entropy			
輸出層 神經元數量	2			

第四章 實驗結果

此章節實驗使用了三個不同遞歸神經網路，RNN、LSTM和GRU，針對不同結構、訓練參數及不同處理方式的數據集，來進行訓練及測試，研究哪一項方法對跌倒事件的判斷擁有最好的準確度。

4.1 RNN、LSTM及GRU模型訓練比較

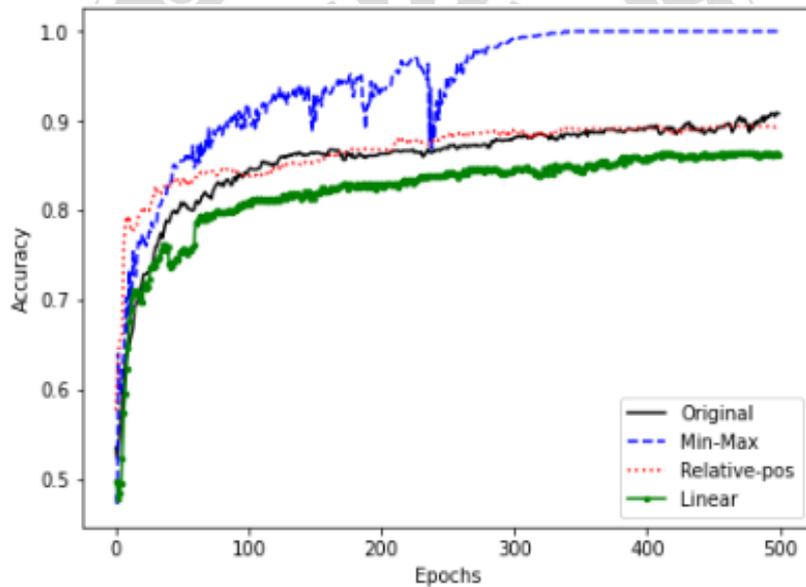
4.1.1 正規化方法比較

此小節針對不同正規化方法做比較，包含未正規化、最大最小值正規化、相對位置正規化及線性插值加相對位置的正規化方法，本文設置隱藏神經元為64，學習率0.01，觀察數據集訓練及驗證曲線。

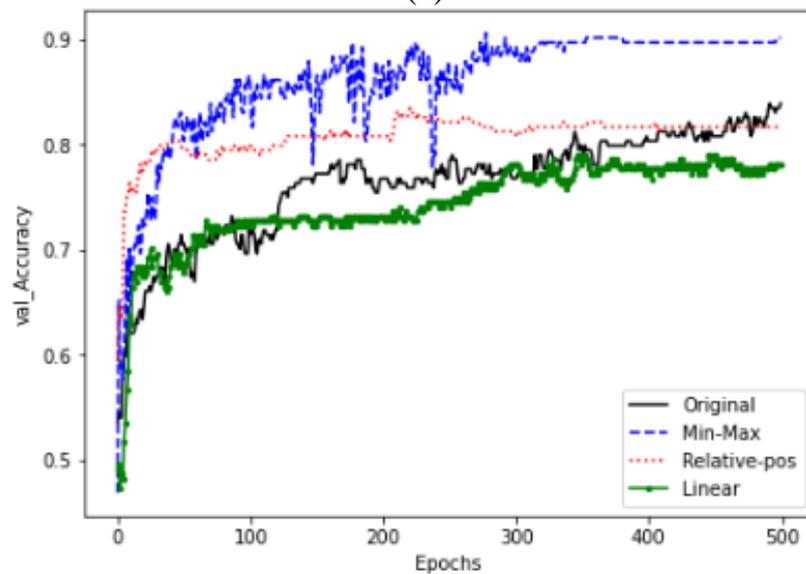
如圖 4-1、RNN模型訓練及驗證準確度(a)所示，能看到RNN在訓練時，最大最小值的方法確實有收斂快速並提高準確度的功能，如圖 4-1、RNN模型訓練及驗證準確度(b)所示，可以看到RNN在驗證時，最大最小值的方法雖然有較高的準確度，但是在50 Epochs到300 Epochs間出現多次大幅度震盪，可能的原因是因為數據集被分為有遺失值和無遺失值，造成數據分佈被改變，導致訓練難度提高。

如圖 4-2、RNN模型訓練及驗證損失所示，圖(a)為RNN訓練損失，能看到最大最小值收斂速度相當快速，在Epochs 350時已經收斂，其他方法還未收斂，還有能下降的空間，但是需要更大的Epochs而訓練時間也會隨之上升，看到圖(b)驗證損失，最大最小值雖然收斂效果優於其他方法，但是在Epochs 100到300間有多次大幅度震盪，之後出現過擬合的現象，發生數據學習過度，只會對訓練集有良好的效果，更換至其他數據集，像是

驗證資料集時就出現大量誤差。



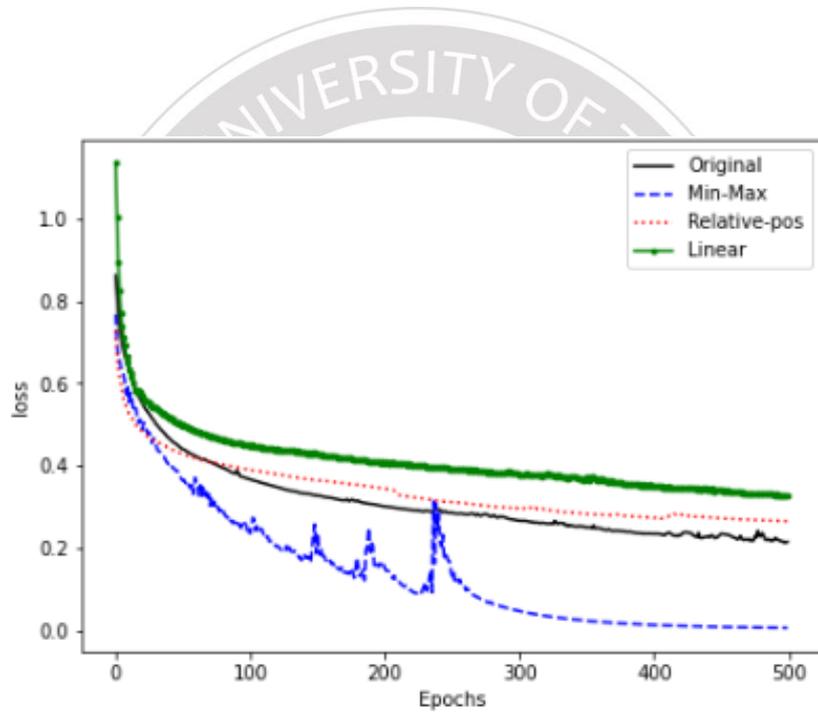
(a)



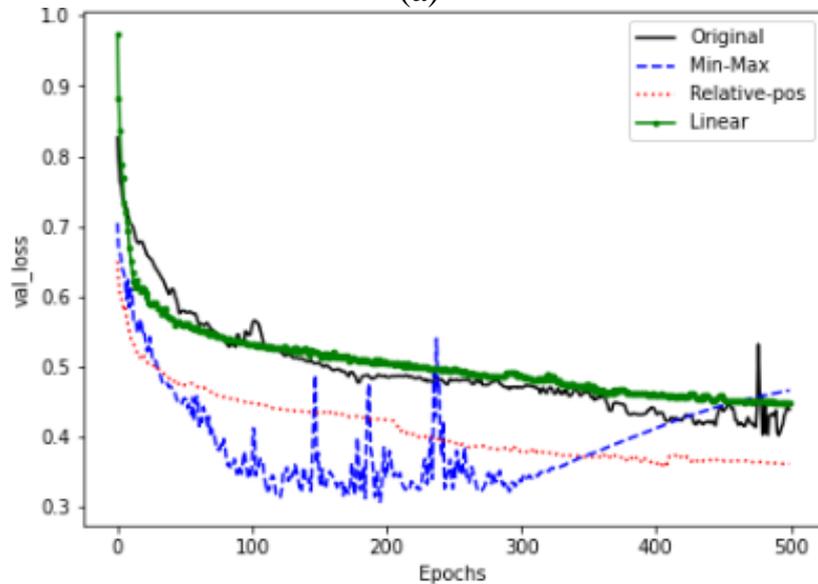
(b)

圖 4-1、RNN模型訓練及驗證準確度(a)訓練準確度 (b)驗證準確度

考量了使用RNN模型的緣故，因為RNN模型結構較簡易，因此針對數據集在較小值區間能有快速收斂的效果，而未正規化、相對位置和線性插值加相對位置方法，數據集為圖片大小為區間，所以收斂速度較慢，改善的方法可以從結構和參數調整。



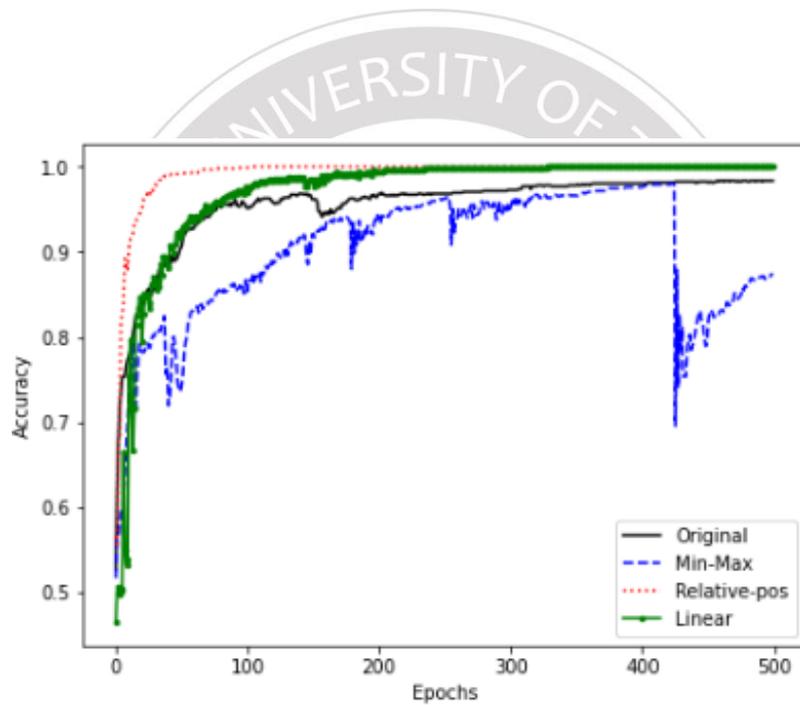
(a)



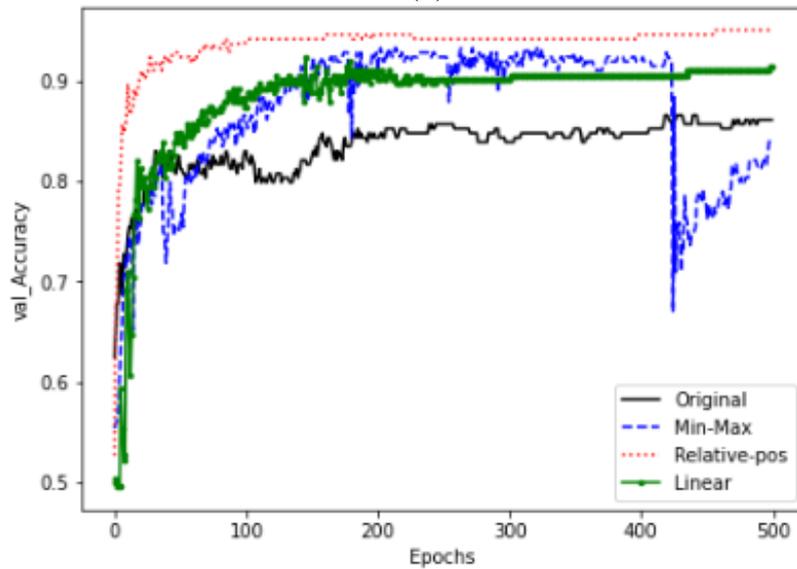
(b)

圖 4-2、RNN模型訓練及驗證損失(a)訓練損失 (b)驗證損失

如圖 4-3所示，圖(a)為LSTM訓練階段，可以看到我們提出的相對位置正規化方法，收斂速度快，準確度較高，而最大最小值在Epochs 420時出現大幅度衰減，原因同樣是訓練數據分佈被破壞，圖(b)為驗證階段，相對位置正規化方法，改善了最大最小值的大幅度震盪的問題，雖然仍然有些微震盪，原因是依然還有遺失點的存在。



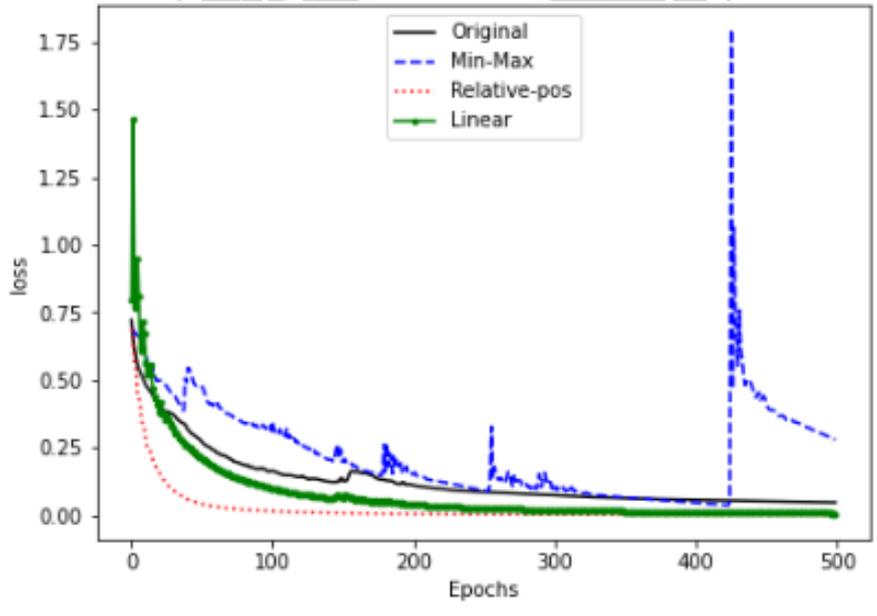
(a)



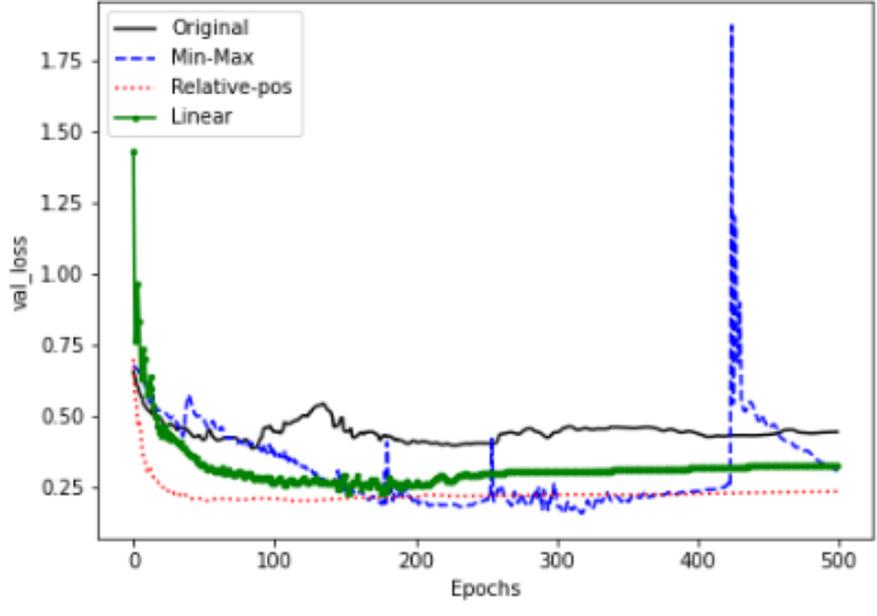
(b)

圖 4-3、LSTM模型訓練及驗證準確度(a)訓練準確度 (b)驗證準確度

如圖 4-4所示，圖(a)為訓練階段損失，LSTM的結構改善了RNN長期記憶的問題，因此有更好的效能，相對位置正規化方法在Epochs 50時就進入收斂，而線性插值加上相對位置方法，在Epochs 250時進入收斂，並且損失誤差下降到小於0.25，圖(b)為驗證階段，發現我們提出的方法包含使用線性插值，出現了過擬合的狀況，原因是結構或是學習參數不合適。

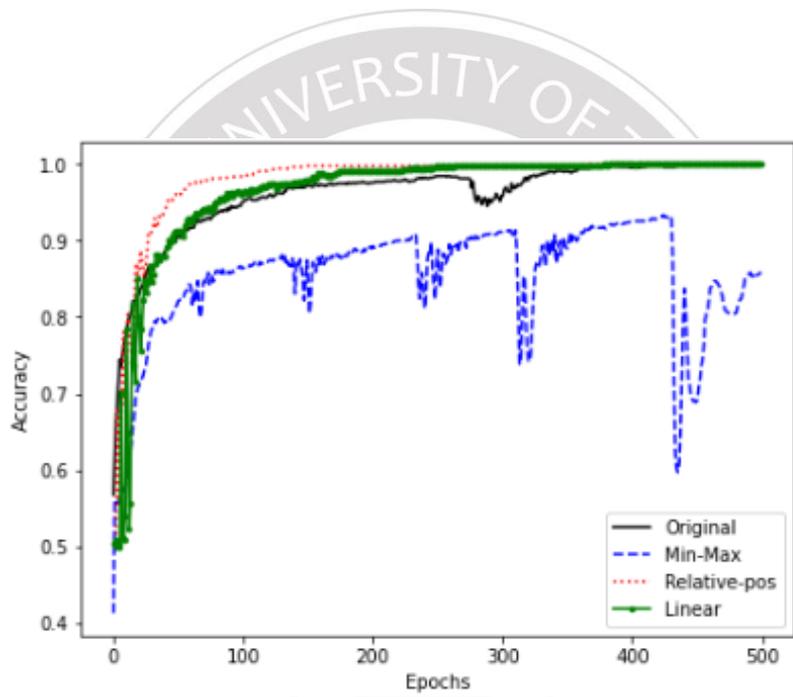


(a)

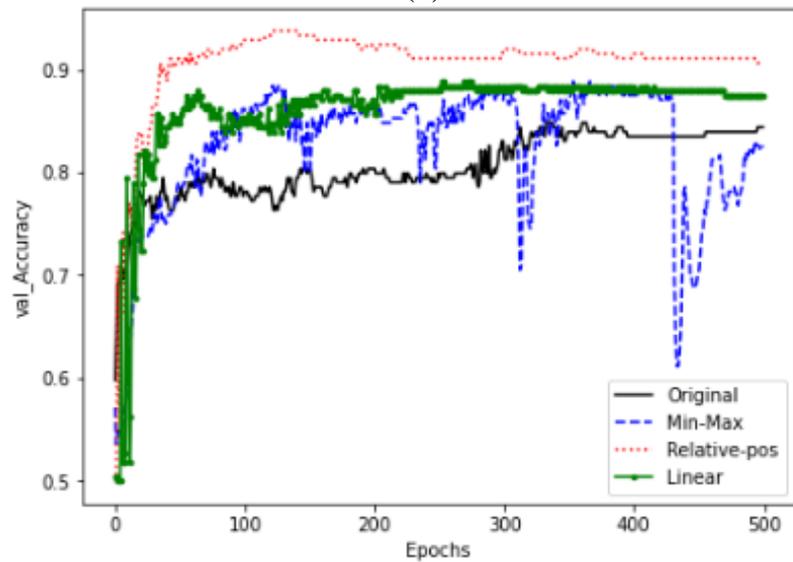


(b)

圖 4-4、LSTM模型訓練及驗證損失(a)訓練損失 (b)驗證損失



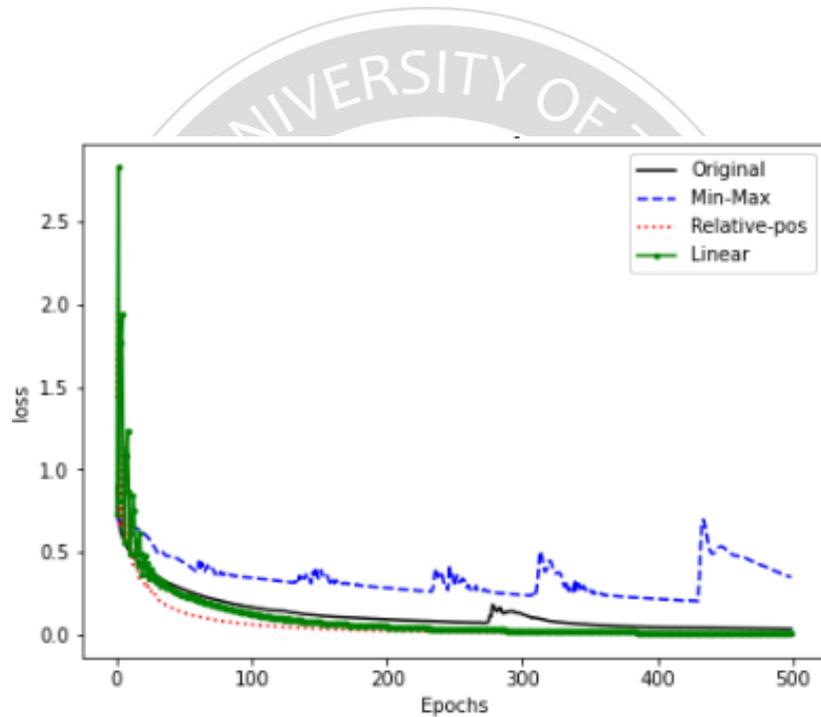
(a)



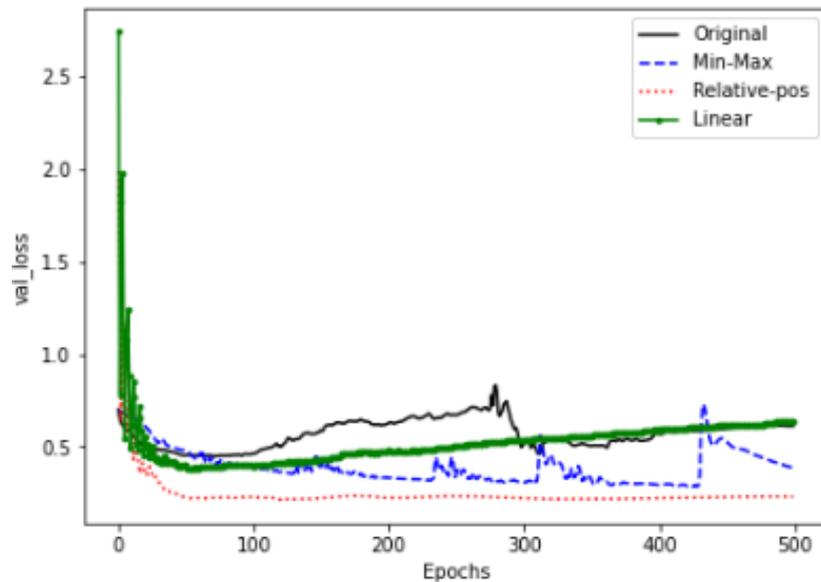
(b)

圖 4-5、GRU模型訓練及驗證準確度(a)訓練準確度 (b)驗證準確度

如圖 4-6所示，圖(a)為訓練損失，除了最大最小值方法外，其他3種方法都有較低的誤差損失，而相對位置正規化，是收斂最快速的，在 Epochs 50時就進入收斂，圖(b)為驗證階段，相對位置正規化是效果最佳的，驗證誤差損失最小也最平穩，加上線性插值後，出現了過擬合的問題，因此本文認為是結構或是訓練參數不合適。



(a)



(b)

圖 4-6、GRU模型訓練及驗證損失(a)訓練損失 (b)驗證損失

如圖 4-7所示，各個數據集在隱藏神經元為64，學習率為0.01的情況下，未正規化的數據集的驗證階段，三個模型的有震盪並出現過擬合的情況，在驗證資料評估下RNN平均準確度為77.6%、LSTM為87%而GRU為82%。最大最小值不論訓練還是驗證階段，雖然準確度有持續上升，但都出現大幅度震盪並不穩定，我們認為是數據集的分佈被改變導致增加了不

必要的特徵，像是有遺失值和無遺失值的分佈情況，提高了學習的困難度，在驗證資料評估下RNN平均準確度為86.6%、LSTM為93%而GRU為83%，準確度之所以上升，可能的原因是最大最小值能增加收斂速度。而使用相對位置及線性插值，都有明顯的改善過擬合的問題，準確度分別為82.6%、96%、93%和78.6%、94.3%、92%，線性插值的準確度出現下降，認為是補償值誤差所造成特徵被消除。

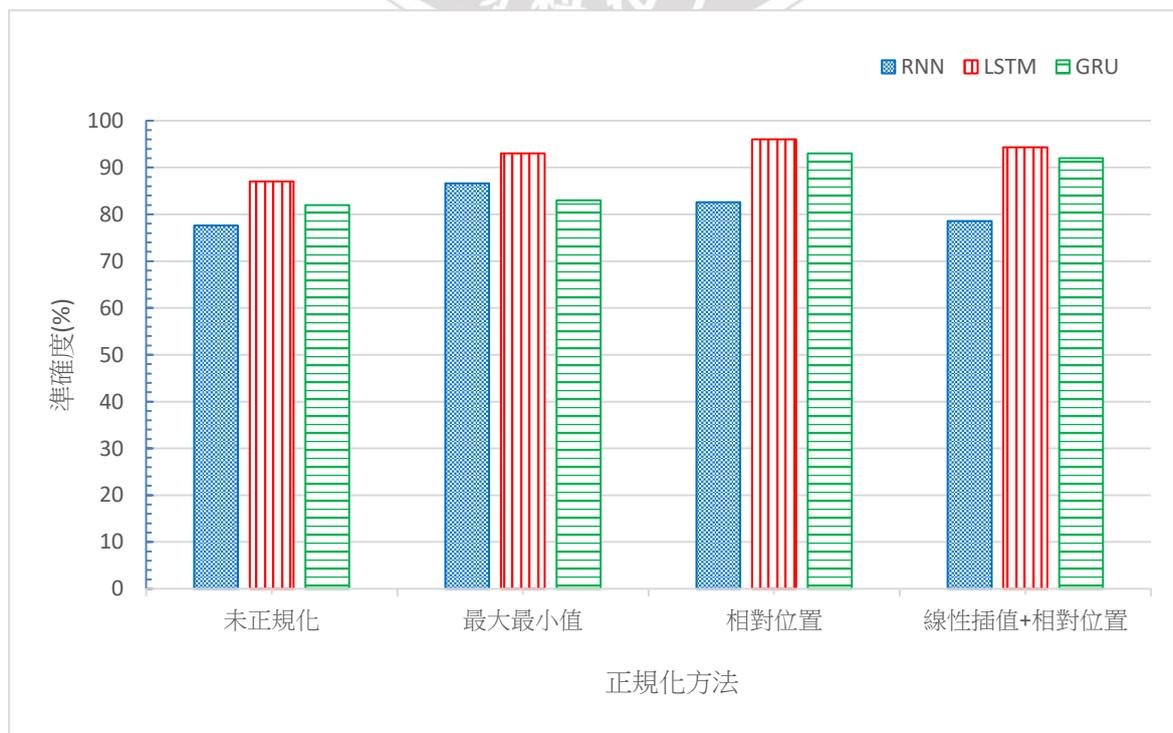


圖 4-7、不同正規化方法平均準確度比較

4.1.2 隱藏神經元比較

此小節將測試不同的隱藏神經元(Hidden Node)對相對位置正規化數據集的影響，分別有64、128、256、512和1024個不同的數量，學習率為0.01，若神經元太少，可能會無法描述問題；當神經元越多能找到更小的誤差，但收斂會越慢，收斂速度是指到達最小誤差的時間。可是若一直不

斷增加神經元數量，將會增加運算時間，且不一定能找到更小的誤差。

如圖 4-8所示，能看到64個神經元收斂速度是最慢的，在Epochs 40後進入平穩，正常情況下，64神經元應該是收斂速度最快的，原因是RNN結構較簡單，當模型複雜度低時，會導致無法完整描述問題，因此無法得到最小誤差損失。而找到最小誤差的神經元數量為1024及512，誤差損失下降至小於0.4。

如表 4-1所示，64神經元的準確度是相對較低的，僅有82.6%，證實無法找到最小誤差導致準確度下降，並且收斂異常。看到1024神經元準確度為88%，是RNN模型表現較好的神經元數量，因結構簡單所以需要較多神經元來改善學習效能。

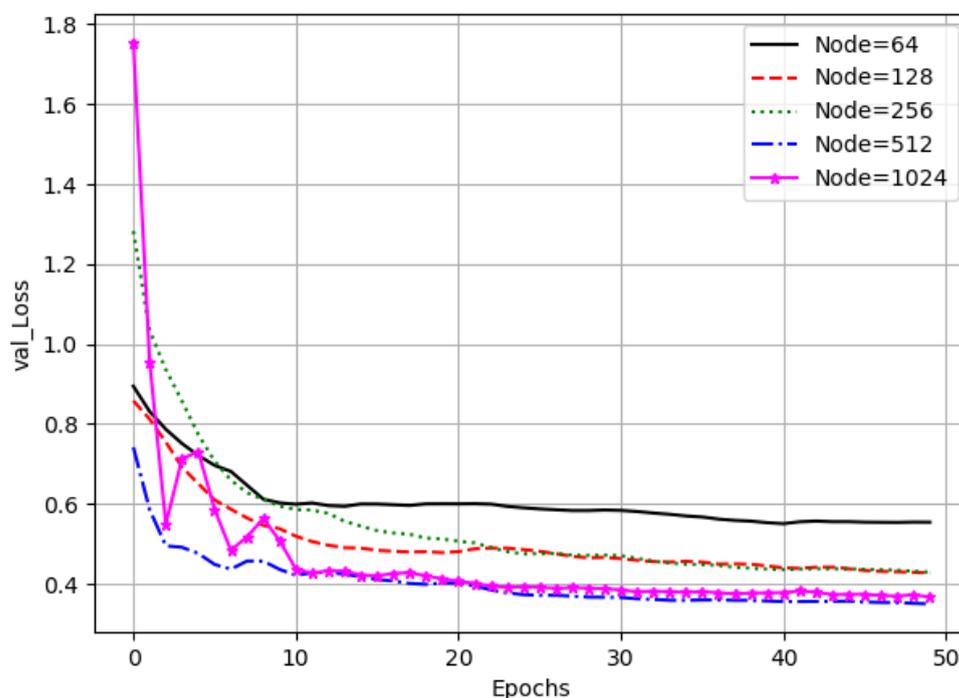


圖 4-8、RNN不同神經元數量之收斂速度比較

表 4-1、RNN隱藏神經元平均準確度

隱藏神經元數量	平均準確度(Acc)
64	82.6%
128	84%
256	86.3%
512	87.3%
1024	88%

如圖 4-9、LSTM不同神經元數量之收斂速度比較所示，LSTM是相當強大的模型，因結構相當複雜且規則嚴苛，所以不論神經元數量多寡，都有亮眼的表現，能看到1024神經元因數量最多，因此收斂速度最慢；而64神經元收斂速度最快。

如表 4-2所示，1024神經元對於LSTM而言並沒有太好的效果，準確度94.6%，在512時卻得到最高的準確度98.3%，我們認為原因是LSTM模型結構複雜，當使用過多神經元時，會造成問題被過度描述，因此造成準確度下降。

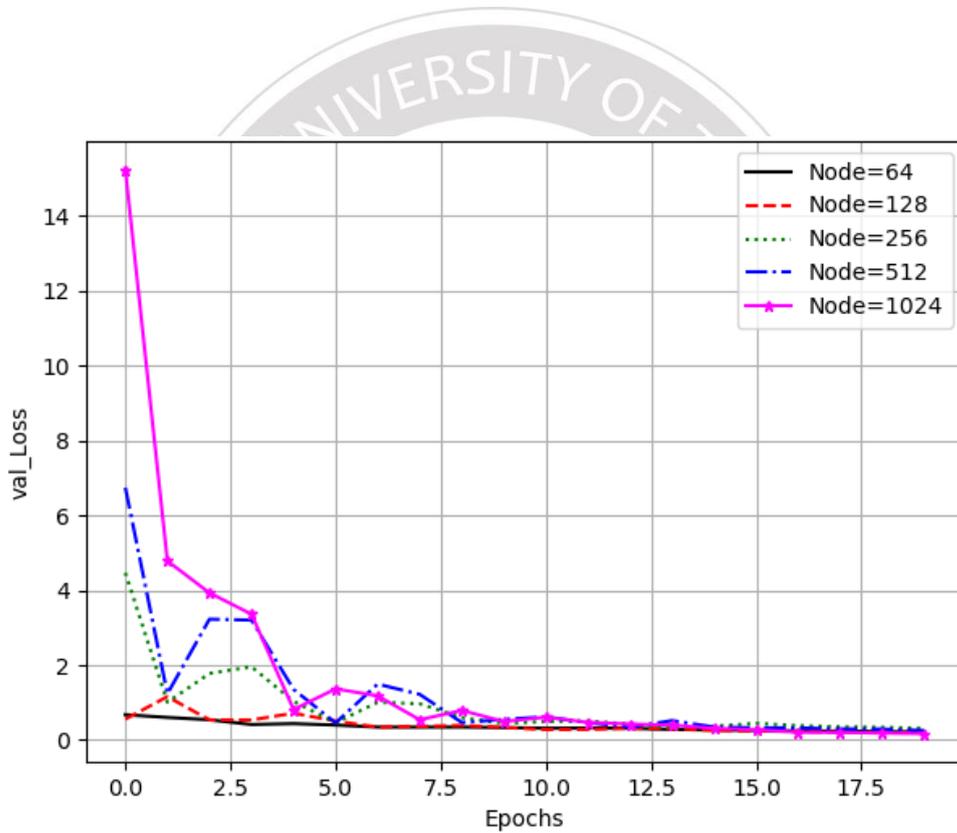


圖 4-9、LSTM不同神經元數量之收斂速度比較

表 4-2、LSTM隱藏神經元平均準確度

隱藏神經元數量	平均準確度(Acc)
64	96%
128	96%
256	97.3%
512	98.3%
1024	94.6%

如圖 4-10所示，GRU的各神經元收斂速度也是正常情況，當神經元越多表示需要的問題描述越多，因此收斂速度下降，而神經元過少時，會發生問題不夠描述的問題無法區分數據分佈，但收斂速度較快，精準度低。

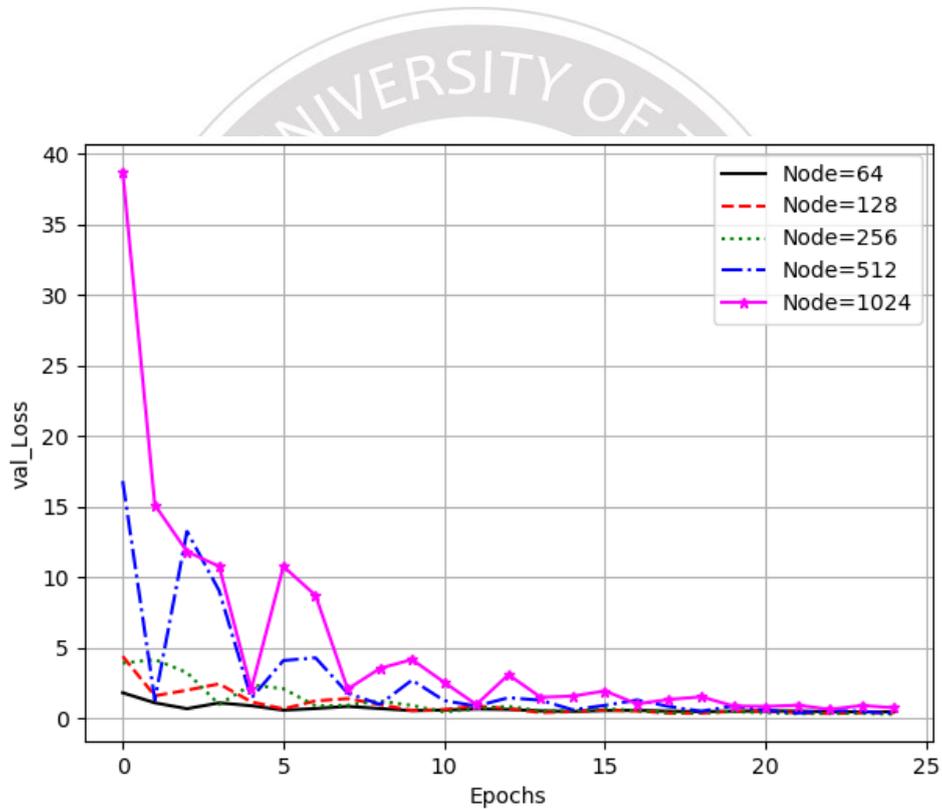


圖 4-10、GRU不同神經元數量之收斂速度比較

表 4-3、GRU隱藏神經元平均準確度所示，GRU和LSTM有著相同的情況，雖然結構比起LSTM較簡易，但性能卻不比LSTM差很多，在512神經元得到最高準確度96.6%。

表 4-3、GRU隱藏神經元平均準確度

隱藏神經元數量	平均準確度(Acc)
64	93%
128	94%
256	96%
512	96.6%
1024	94.6%

4.1.3 學習率比較

此小節選擇線性插值加相對位置和相對位置數據集做比較，因為在訓練和驗證階段的Loss出現些微過擬合的問題。我們在神經元為512的情況

下，測試了0.1、0.01和0.001的學習率找出最合適的學習參數。

如圖 4-11所示，能看到RNN在學習率0.1的情況下，找到了最高準確度，準確度為89%，而LSTM和GRU在學習率0.01的情況下，找到了較高準確度，準確度分別為98.3%及96.6%。

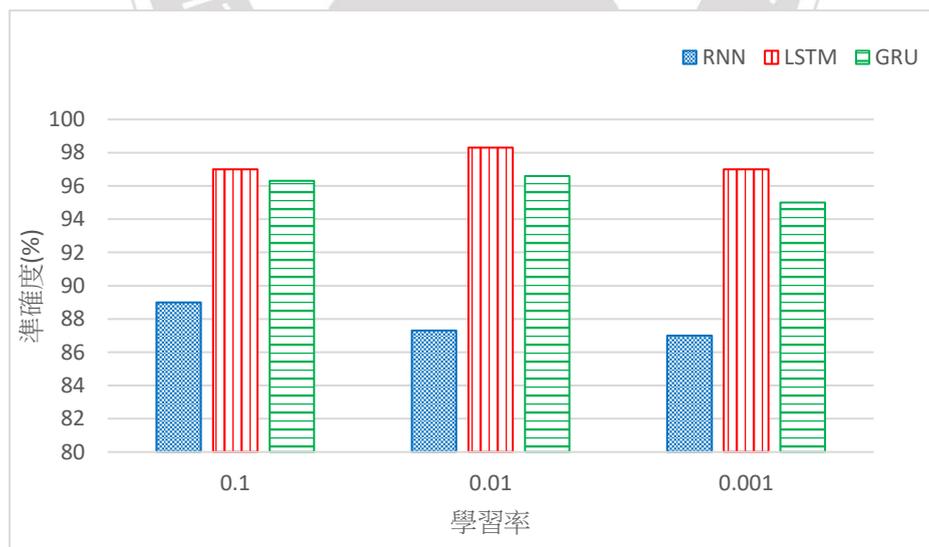


圖 4-11、不同學習率之平均準確度比較(相對位置)

如圖 4-12所示，能看到三個模型都在學習率0.1時找到最高的準確度，準確度分別為88.63%、93.6%和96.3%，發現經過插值後準確度沒有改善，可能是插值方法並不能對遺失值有正確的補償效果。

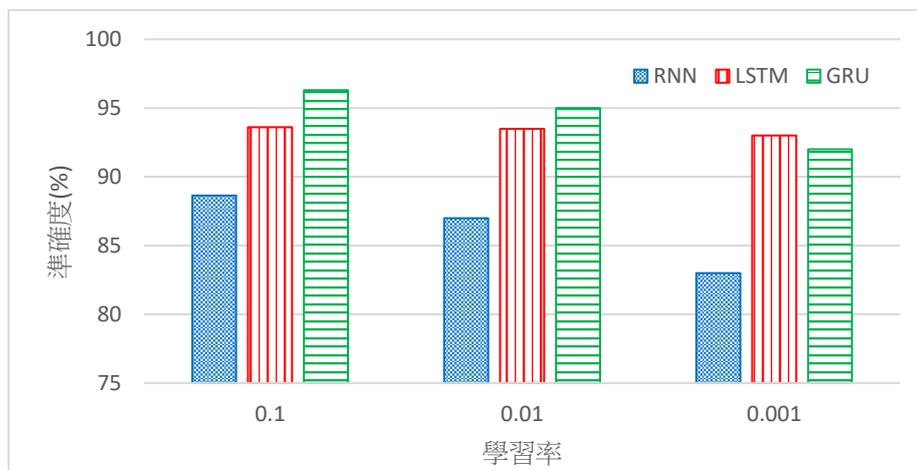


圖 4-12、不同學習率之平均準確度比較(線性插值)

4.2 RNN、LSTM及GRU模型評估指標

此小節將討論每個數據集的三個模型的混淆矩陣，能看到有真實事件和模型預測，分為四種預測情況：(True Positive, TP)、(True Negative, TN)、(False Positive, FP)和(False Negative, FN)，使用這四個參數，去計算出模型的評估指標。我們分配了跌倒事件112件，非跌倒事件112件，共有224個事件讓模型進行預測。模型的結構及參數是使用以上小節獲得較高準確度的結構和參數，未正規化、最大最小值及相對位置是採用512個隱藏神經元及0.01學習率，而線性插值為0.1學習率。

如圖 4-13所示，RNN 模型針對4種不同正規化方法做訓練，看到圖(b)中的FN和FP是最高的，表示跌倒和非跌倒事件檢測錯誤的數量，因為最大最小值將遺失值一併計算，導致無遺失值的連續動作而出現特徵差異。看到圖(c)(d)中的FN和FP，都比圖(a)未正規化的檢測錯誤數量還少，因此我們提出的做法使有效的。

如圖 4-14所示，靈敏度表示正確分類跌倒事件，特異度表示正確分類非跌倒事件，因此靈敏度越高表示判斷跌倒事件越準確。反之，準確度表示模型正確分類跌倒和非跌倒事件，越高表示正確分類事件的數量越多。可以看到RNN模型整體的準確度最高為90.1%，是使用相對位置加線性插值，雖然提升了特異度但是靈敏度下降了一點，而最高的靈敏度為相對位置正規化方法。

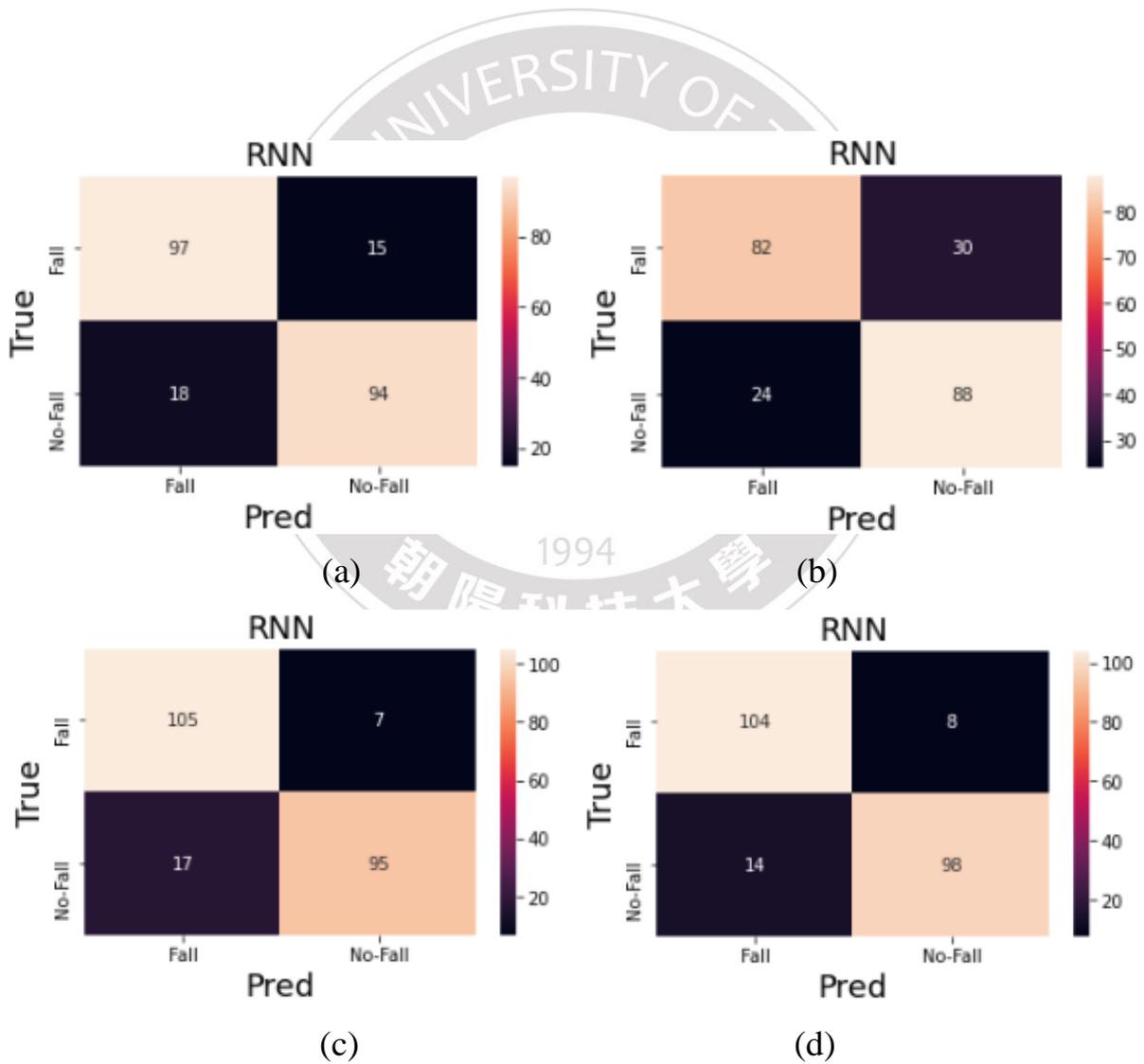


圖 4-13、RNN混淆矩陣(a)未正規化 (b)最大最小值 (c)相對位置 (d)相對位置+線性插值

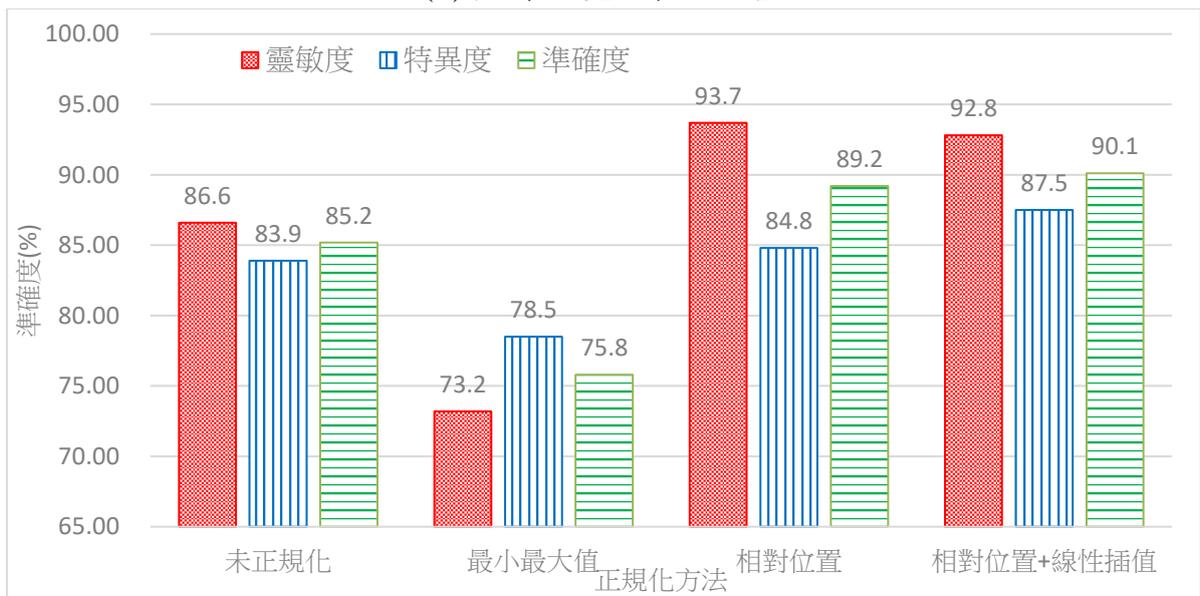


圖 4-14、RNN模型正規化方法評估指標比較

如圖 4-15所示，可以看出圖(b)使用了最大最小值後比未正規化的FP和FN的數量來的多，整體效果都下降，看到圖(c)使用相對位置方法整體效果上升許多，FN為0、FP為4，誤差比起其他方法都低，而圖(d)是使用線性插值補償缺失值後，誤差並沒有再下降，原因是補償缺失值的方法，可能會導致原始擁有的特徵被消除。

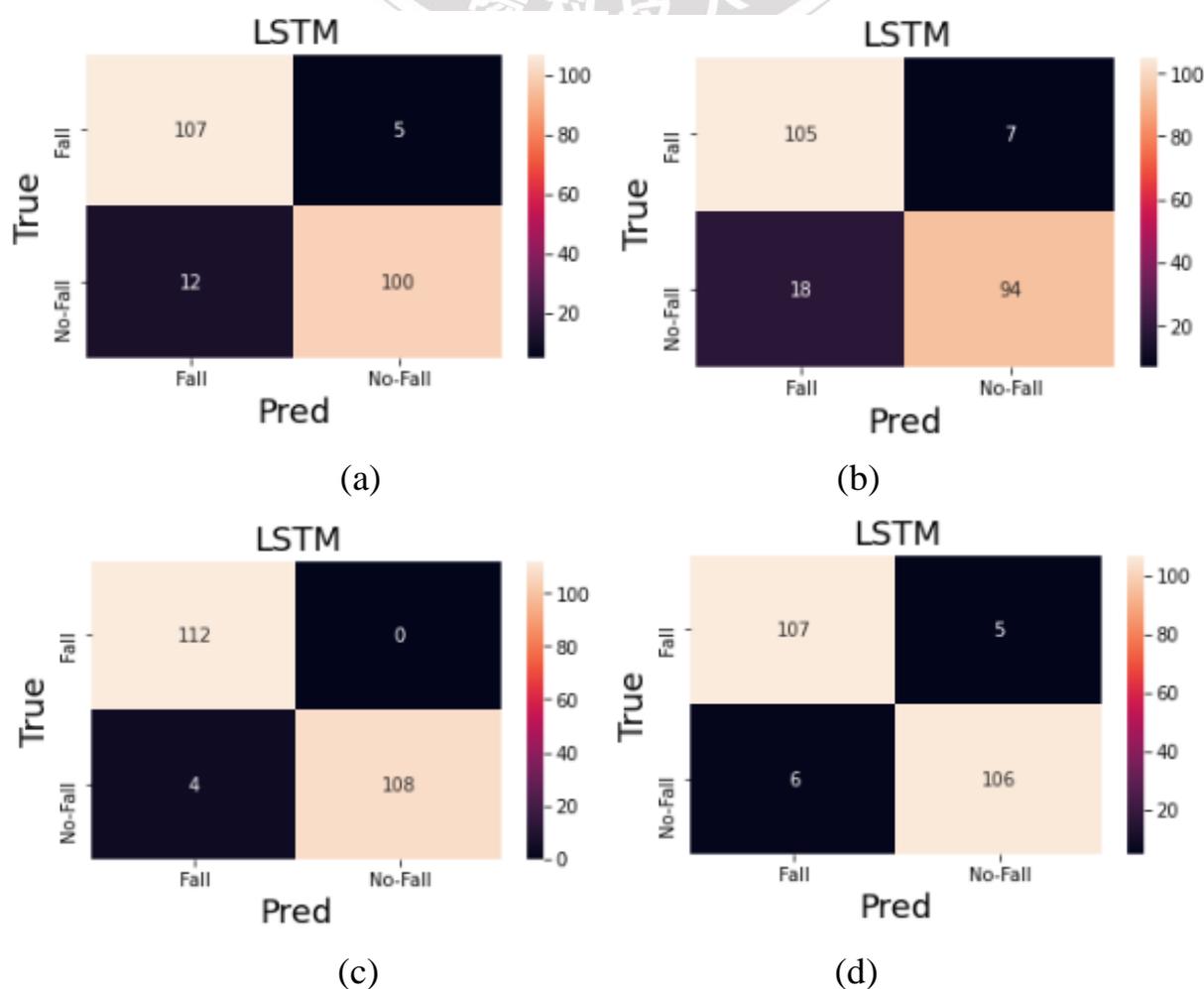


圖 4-15、LSTM混淆矩陣(a)未正規化 (b)最大最小值 (c)相對位置 (d)相對位置+線性插值

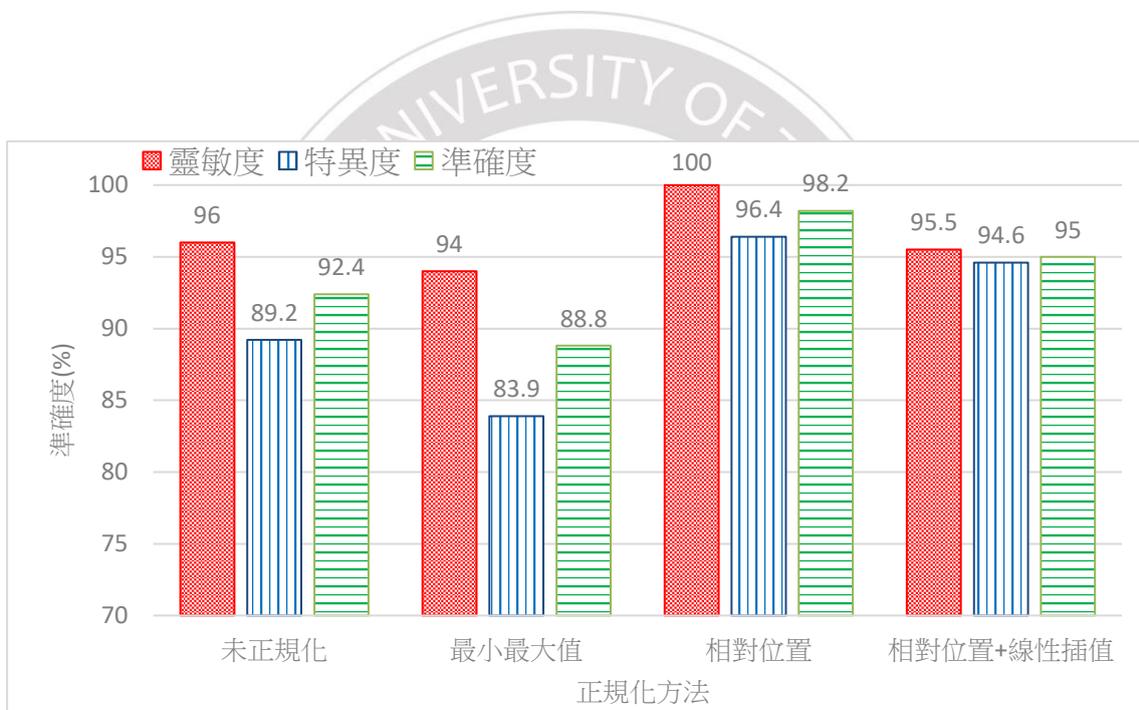


圖 4-16、LSTM模型正規化方法評估指標比較

如圖 4-17所示，圖(a)為GRU模型未正規化混淆矩陣，發現FP比LSTM和RNN還要多，表示非跌倒檢測錯誤的事件變多，GRU為了改善LSTM構造複雜造成訓練時間長和硬體問題，而減少內部閘門，因此有部分特徵的限制比LSTM來的寬鬆，圖(c)為使用相對位置正規化，FP得到巨大的改善，從26減少到2個事件，GRU模型也是三個模型中檢測非跌倒事件準確度較高的。

如圖 4-18所示，GRU模型因減少結構所以特徵篩選條件較寬鬆，因此在未正規化的數據集中，沒有像LSTM一樣有不錯的準確度，但是在使用相對位置正規化後，特異度有明顯的上升，從76.7%上升到98.2%，表示非跌倒事件的分類準確度，整體準確度從85.2%上升到97.3%，雖然GRU準確度沒有LSTM那麼好，但特異度是三個模型中較好的。

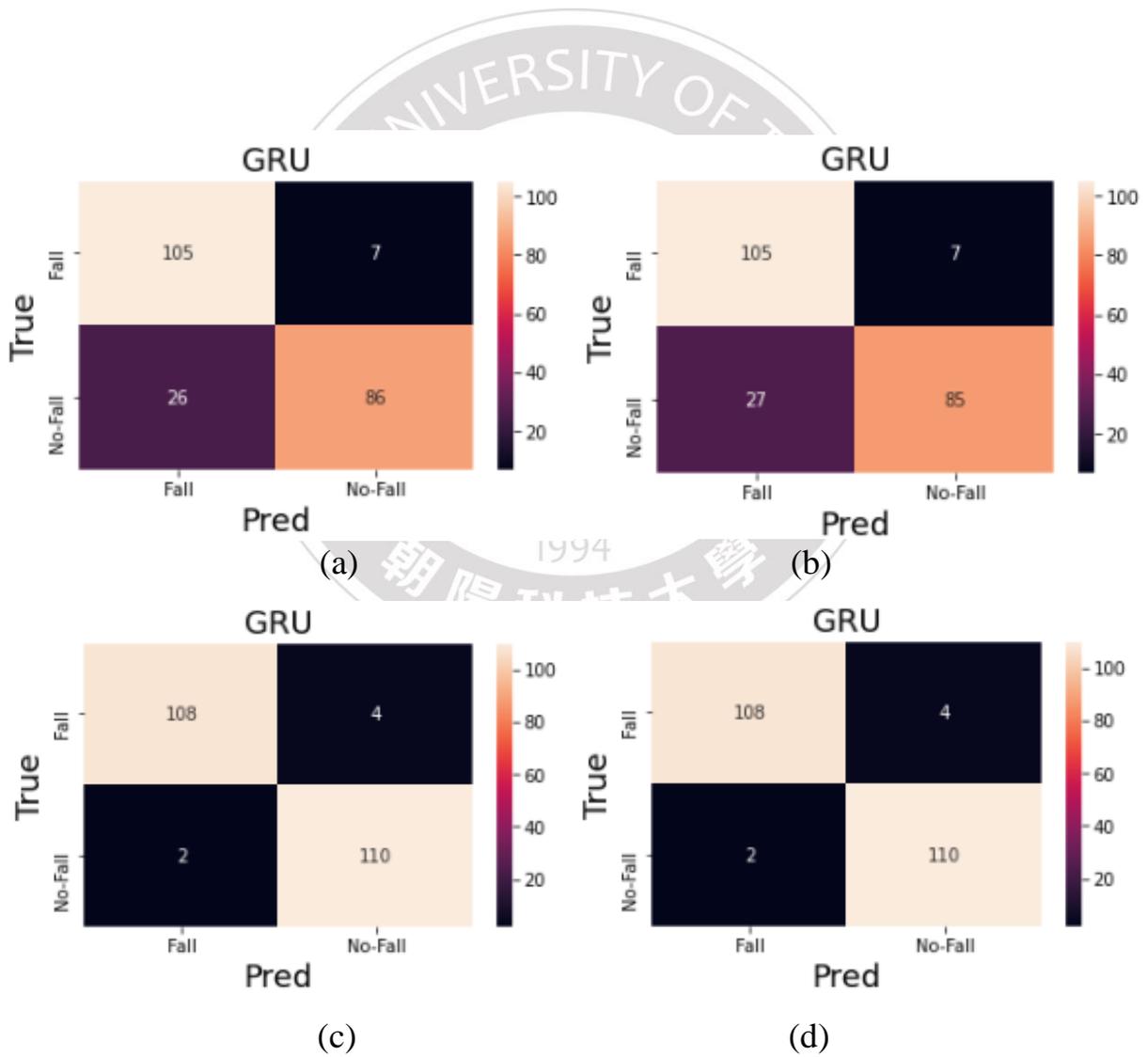


圖 4-17、GRU混淆矩陣(a)未正規化 (b)最大最小值 (c)相對位置 (d)相對位置+線性插值

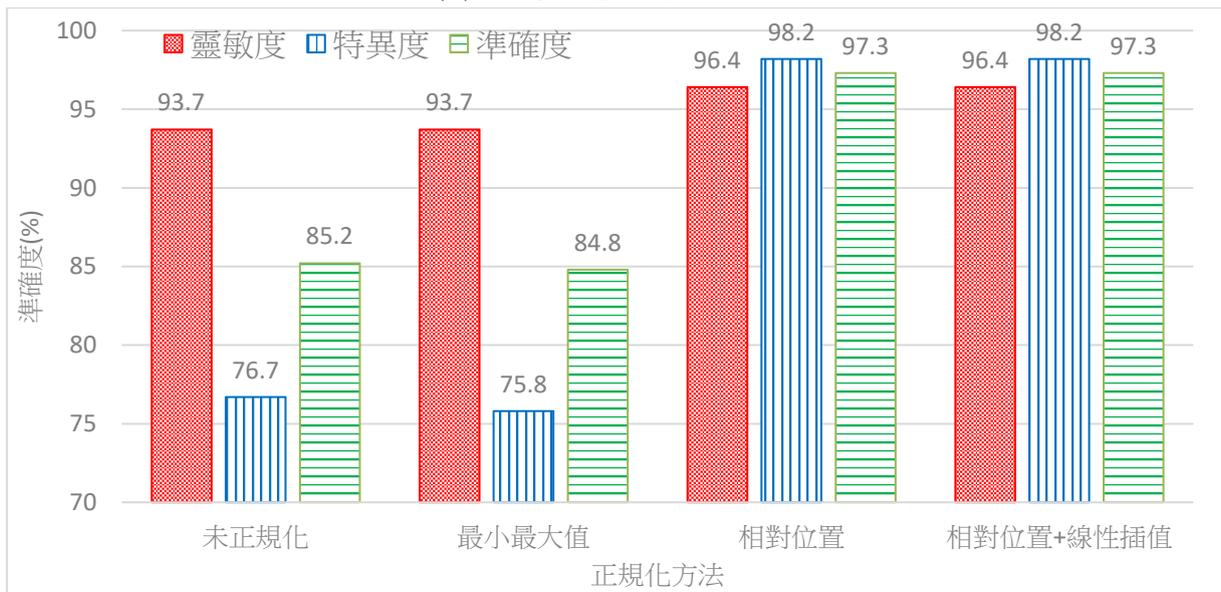


圖 4-18、GRU模型正規化方法評估指標比較

4.3 RNN、LSTM及GRU模型訓練速度比較

4.3.1 硬體比較

如圖 4-19所示，使用CPU來做訓練，RNN的訓練時間是最短的，由於模型結構比起其他兩個模型較為單純。而LSTM模型結構是最複雜的，所以比起其他的耗費更多的時間。GRU是為了改善LSTM的執行速度及耗費能源的問題，因此在圖中能發現LSTM略高於GRU的訓練時間。

看到GPU的訓練時間差距很大，RNN差了2倍訓練時間，LSTM差了約6倍訓練時間而GRU差了約4倍訓練時間，但LSTM略快於GRU的訓練時間，我們認為是因為Google平台的硬體設備，有對LSTM做優化導致訓練速度上升。

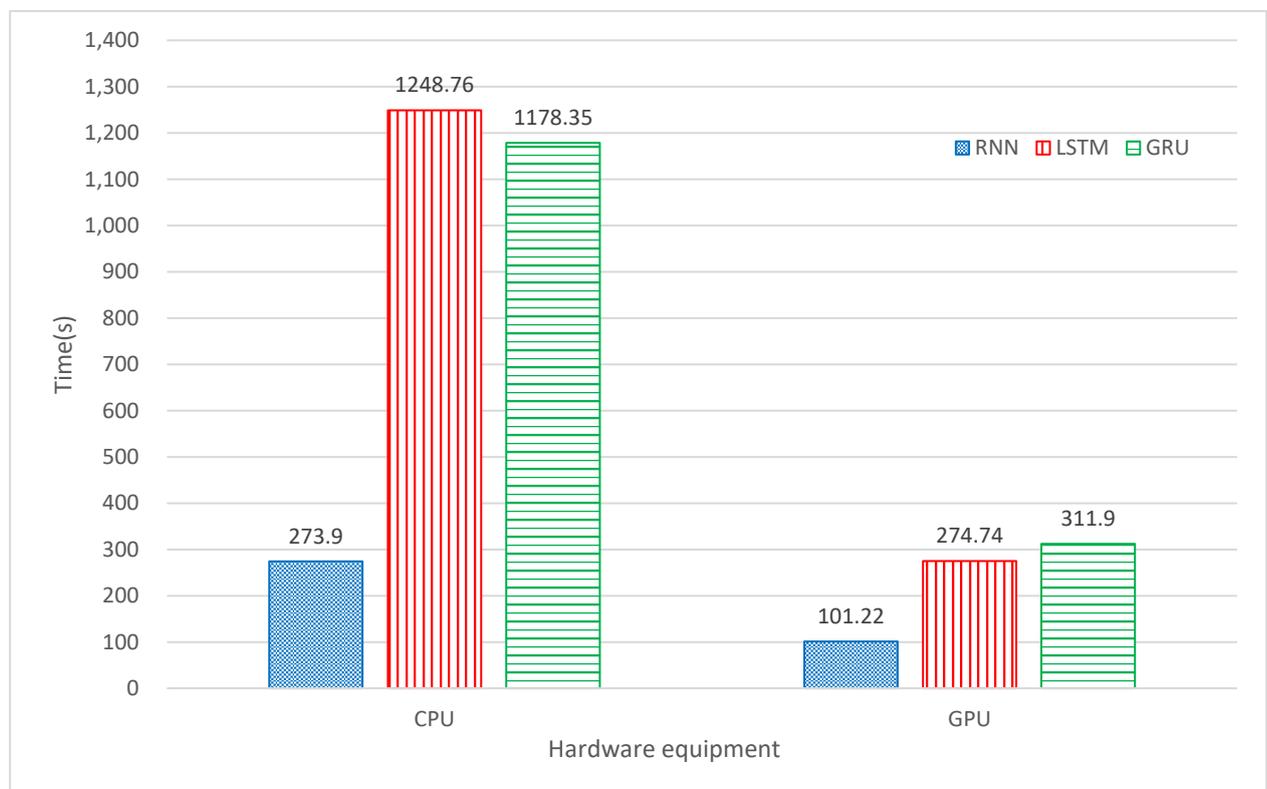


圖 4-19、模型訓練速度

4.3.2 正規化方法比較

如圖 4-20所示，不同正規化方法訓練速度也不同，因為未正規化和最大最小值的輸入為50，所以比起其餘兩個正規化方法有更多數據需要處理，因此訓練時長比較長，RNN約為100秒、LSTM約為275秒和GRU約為311秒，而相對位置和線性插值只有30個輸入，因此訓練時長較短，RNN約為63秒、LSTM約為175秒和GRU約為200秒。

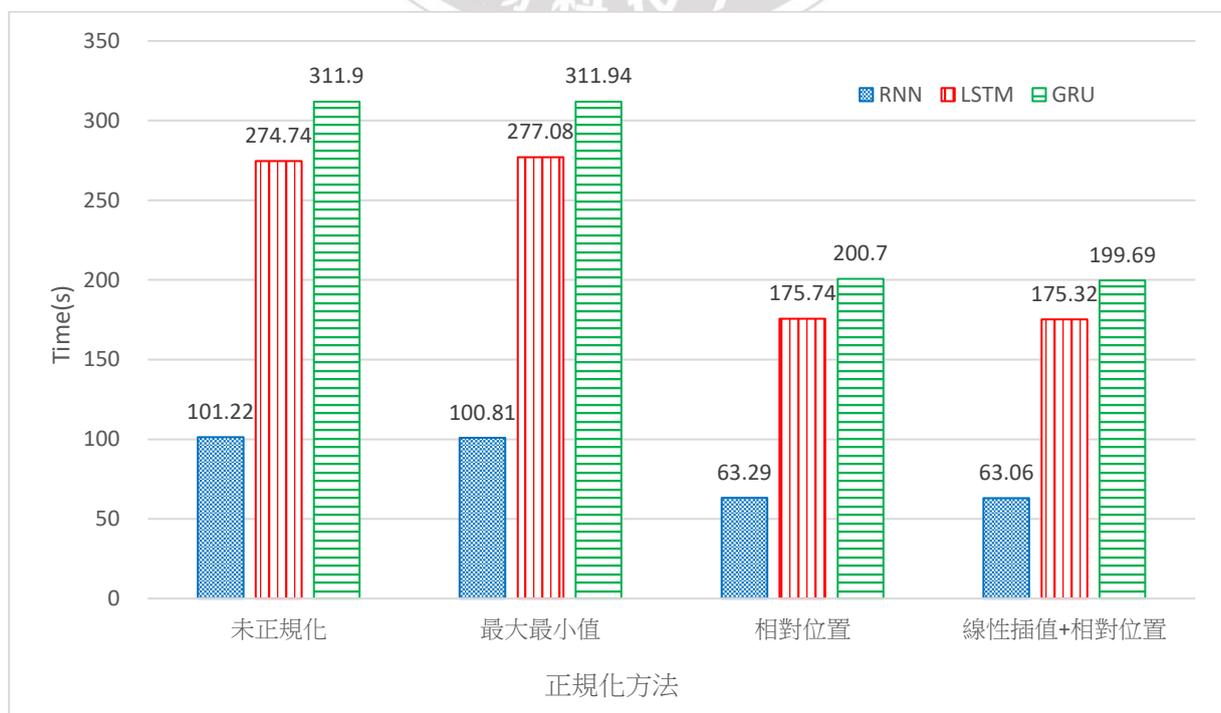


圖 4-20、正規化方法訓練速度比較

4.4 小結

本章節的實驗結果可以發現，最大最小值得處理方式，較不適合運用在本文的數據集，原因是原始數據是從圖像中取得，原本就把數據限制在圖像大小的範圍，若再次做最大最小值，會因為遺失點導致特徵消失。而未正規化的驗證情況不佳，是因為噪音過多所造成的，相對位置改善了

以上兩者的問題，減少過多的特徵和噪音，沒有把遺失點一併計算，因此得到不錯的結果，而線性插值的部分，因為轉折點的緣故導致補償值誤差過大，讓有特徵被影響，準確率下降。

從實驗結果也得知三個遞歸神經網路的成果，在使用相對位置處理的數據集中，RNN因為自身結構較為簡單，因此對長期記憶沒有太好的效果，會因為反向傳播更新權重造成梯度爆炸和消失。而LSTM和GRU改善了這項缺點，使用閘門來更新記憶，但是LSTM擁有4個閘門結構複雜，所以訓練時間也是最長的，相反的，GRU改善了LSTM的缺點，讓結構變得較簡易，訓練時間也變快，也擁有與LSTM相等的準確度。

三個模型的準確度為89.2%、98.2%和97.3%，由LSTM取得最高的準確度，能夠最好辨識跌倒的發生，其次是GRU，最後是RNN，由於GRU的規則條件還是比LSTM寬鬆，因此在某些記憶的選擇還是比LSTM遜色，但GRU更適合大型並特徵較少的數據集，會有更好的效果。

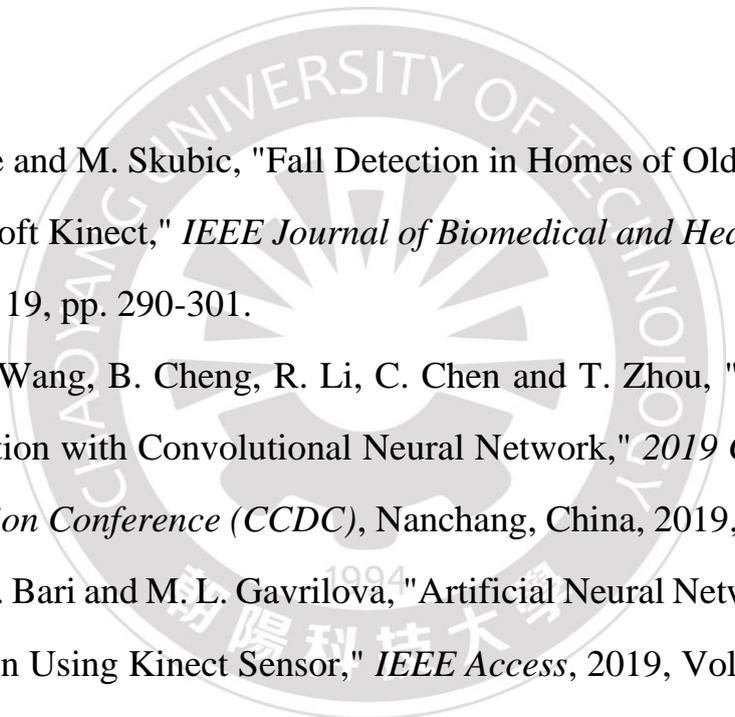
第五章 結論與未來工作

本文提出一種使用遞歸神經網路對於人體骨架姿勢的辨識方法。使用OpenPose來取得人體骨架，因人體骨架有關連性，我們認為關節點的變化軌跡是與人體動作有所關聯。此方法除了在複雜的空間環境下有不錯效果，還能減少設備成本，不需要過多的感測器也不會造成年長者有穿戴上的困擾。但關節點會在某些姿勢動作情況下遺失，造成模型在訓練時有過擬合的現象，因此我們將關節點位移到設置的相對位置，與部分數據插值來改善噪音過多和未正規化的情況。遞歸神經網路可以有效地幫助學習人體關節點的時間變化規律，並減少訓練時間，在實驗結果中，RNN準確率為89.2%、LSTM為98.2%和GRU為97.3%，因此本文提出跌倒檢測法之準確率相當高。

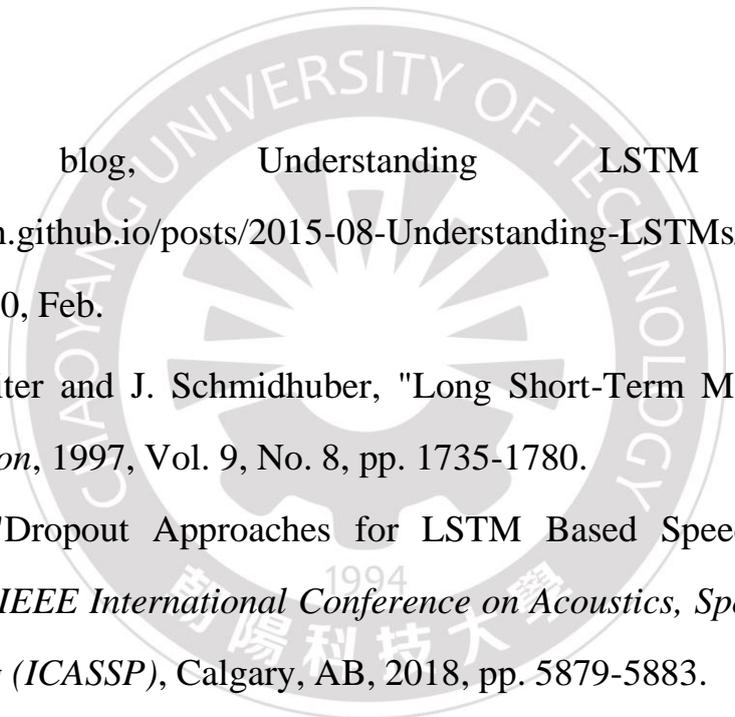
未來將嘗試在關節點遺失的補償和動作預測等工作，本文目前是使用線性的插值方法，除了改善擬合情況，也發現線性插值方法的缺陷，因此將嘗試使用遞歸神經網路，做關節點的預測遺失點以提升辨識準確率。再來是嘗試動作預測的工作，若能提前預知就能快速的得到跌倒的訊息，將可以防止或更快發現跌倒事件。

參考文獻

- [1] 國家發展委員會, 中華民國人口推估(2018至2065年), 國家發展委員會, 2018, 第1-1 - 2-11頁。
- [2] United Nations, Department of Economic and Social Affairs, Population Division. World Population Prospects 2019: Highlights, (ST/ESA/SER.A/423), 2019
- [3] United Nations, Department of Economic and Social Affairs, World Population Prospects The 2015 Revision, (ESA/P/WP.241), 2015
- [4] 衛生福利部國民健康署, 國民健康訪問調查, <https://www.hpa.gov.tw/Pages/Detail.aspx?nodeid=807&pid=4326>, 查詢日期: 2020年3月
- [5] 世界衛生組織, 跌傷, <https://www.who.int/zh/news-room/factsheets/detail/falls>, 查詢日期: 2020年3月
- [6] Y. Nho, J. G. Lim and D. Kwon, "Cluster-Analysis-Based User-Adaptive Fall Detection Using Fusion of Heart Rate Sensor and Accelerometer in a Wearable Device," *IEEE Access*, 2020, Vol. 8, pp. 40389-40401.
- [7] F. Hussain, F. Hussain, M. Ehatisham-ul-Haq and M. A. Azam, "Activity-Aware Fall Detection and Recognition Based on Wearable Sensors," *IEEE Sensors Journal*, 2019, Vol. 19, pp. 4528-4536.
- [8] C. Taramasco et al., "A Novel Monitoring System for Fall Detection in Older People," *IEEE Access*, 2018, Vol. 6, pp. 43563-43574.
- [9] W. Lie, A. T. Le and G. Lin, "Human Fall-Down Event Detection Based on 2D Skeletons and Deep Learning Approach," *2018 International Workshop on Advanced Image Technology (IWAIT)*, Chiang Mai, 2018, pp. 1-4.

- 
- [10] E. E. Stone and M. Skubic, "Fall Detection in Homes of Older Adults Using the Microsoft Kinect," *IEEE Journal of Biomedical and Health Informatics*, 2015, Vol. 19, pp. 290-301.
- [11] J. Wu, K. Wang, B. Cheng, R. Li, C. Chen and T. Zhou, "Skeleton Based Fall Detection with Convolutional Neural Network," *2019 Chinese Control And Decision Conference (CCDC)*, Nanchang, China, 2019, pp. 5266-5271.
- [12] A. S. M. H. Bari and M. L. Gavrilova, "Artificial Neural Network Based Gait Recognition Using Kinect Sensor," *IEEE Access*, 2019, Vol. 7, pp. 162708-162722.
- [13] Z. Cao, T. Simon, S. Wei and Y. Sheikh, "Realtime multi-person 2D Pose Estimation Using Part Affinity Fields," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, 2017, pp. 1302-1310.
- [14] A. Viswakumar, V. Rajagopalan, T. Ray and C. Parimi, "Human Gait Analysis Using OpenPose," *2019 Fifth International Conference on Image Information Processing (ICIIP)*, Shimla, India, 2019, pp. 310-314.
- [15] J. Chang and C. Tseng, "Analysis of Correlation Between Secondary PM2.5 and Factory Pollution Sources by Using ANN and the Correlation Coefficient," *IEEE Access*, 2017, Vol. 5, pp.22812-22822.
- [16] G. A. Bhatt and P. R. Gandhi, "Statistical and ANN Based Prediction of Wind Power with Uncertainty," *2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI)*, Tirunelveli, India, 2019, pp. 622-627.
- [17] M. Askari and F. Keynia, "Mid-Term Electricity Load Forecasting by A New Composite Method Based on Optimal Learning MLP Algorithm," *IET Generation, Transmission & Distribution*, 2020, Vol. 14, pp. 845-852.

- [18] Z. Sun, Y. Zhang and Q. Ren, "A Reliable Localization Algorithm Based on Grid Coding and Multi-Layer Perceptron," *IEEE Access*, 2020, Vol. 8, pp. 60979-60989.
- [19] S. P. Mahasagara, A. Alamsyah and B. Rikumahu, "Indonesia infrastructure and consumer stock portfolio prediction using artificial neural network backpropagation," *2017 5th International Conference on Information and Communication Technology (ICoICT7)*, Malacca City, 2017, pp. 1-4.
- [20] Lawrence, Steve et al. "Face Recognition: A Convolutional Neural-Network Approach," *IEEE transactions on neural networks*, 1997, Vol.8, pp.98-113.
- [21] H. Amroun, M. H. Temkit and M. Ammi, "Best Feature for CNN Classification of Human Activity Using IOT Network," *2017 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data(SmartData)*, Exeter, 2017, pp. 943-950.
- [22] CH.Tseng, 初探卷積神經網路，
<https://chtseng.wordpress.com/2017/09/12/%E5%88%9D%E6%8E%A2%E5%8D%B7%E7%A9%8D%E7%A5%9E%E7%B6%93%E7%B6%B2%E8%B7%AF/>，查詢日期：2019年11月
- [23] Mikolov, Tomáš et al. "Recurrent Neural Network Based Language Model," *Eleventh annual conference of the international speech communication association*, 2010, pp. 2877-2880.
- [24] K. Jun, D. Lee, K. Lee, S. Lee and M. S. Kim, "Feature Extraction Using an RNN Autoencoder for Skeleton-Based Abnormal Gait Recognition," *IEEE Access*, 2020, Vol. 8, pp. 19196-19207.

- 
- [25] colah's blog, Understanding LSTM Networks, <http://colah.github.io/posts/2015-08-Understanding-LSTMs/>, Available Date : 2020, Feb.
- [26] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, 1997, Vol. 9, No. 8, pp. 1735-1780.
- [27] J. Billa, "Dropout Approaches for LSTM Based Speech Recognition Systems," *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Calgary, AB, 2018, pp. 5879-5883.
- [28] D. Wei, "Prediction of Stock Price Based on LSTM Neural Network," *2019 International Conference on Artificial Intelligence and Advanced Manufacturing (AIAM)*, Dublin, Ireland, 2019, pp. 544-547.
- [29] H. Xue, D. Q. Huynh and M. Reynolds, "SS-LSTM: A Hierarchical LSTM Model for Pedestrian Trajectory Prediction," *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, Lake Tahoe, NV, 2018 , pp. 1186-1194.
- [30] R. Fu, Z. Zhang and L. Li, "Using LSTM and GRU neural network methods for traffic flow prediction," *2016 31st Youth Academic Annual Conference of Chinese Association of Automation (YAC)*, Wuhan, 2016, pp. 324-328.
- [31] M. Pavithra, K. Saruladha and K. Sathyabama, "GRU Based Deep Learning Model for Prognosis Prediction of Disease Progression," *2019 3rd International Conference on Computing Methodologies and Communication (ICCMC)*, Erode, India, 2019, pp. 840-844.

- [32] Cho, Kyunghyun et al, "Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation," *The 2014 Conference on Empirical Methods on Natural Language Processing(EMNLP)*, Bengio, Yoshua, 2014, pp. 1724–1734.
- [33] Kingma, Diederik P. and Jimmy Ba, "Adam: A method for stochastic optimization," *The 3rd International Conference for Learning Representations*, San Diego, 2015.
- [34] Tommy Huang , 機器學習\統計方法：模型評估-驗證指標 , <https://medium.com/@chih.sheng.huang821/> , 查詢日期：2020年3月
- [35] B. Lin, J. Su, H. Chen and C. Y. Jan, "A Fall Detection System Based on Human Body Silhouette," *2013 Ninth International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, Beijing, 2013, pp. 49-52.
- [36] L. Pishchulin et al., "DeepCut: Joint Subset Partition and Labeling for Multi Person Pose Estimation," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, 2016, pp. 4929-4937.
- [37] Bogdan Kwolek, Michal Kepski, "Human Fall Detection on Embedded Platform Using Depth Maps and Wireless Accelerometer", *Computer Methods and Programs in Biomedicine*, 2014, Vol. 117, p.p. 489-501.
- [38] K. Adhikari, H. Bouchachia and H. Nait-Charif, "Activity recognition for indoor fall detection using convolutional neural network," *2017 Fifteenth IAPR International Conference on Machine Vision Applications (MVA)*, Nagoya, 2017, pp. 81-84.

- [39] Google, Colaboratory, <https://research.google.com/colaboratory/faq.html>, Available Date : 2019, Sep.
- [40] T. Nguyen, M. Cho and T. Lee, "Automatic Fall Detection Using Wearable Biomedical Signal Measurement Terminal," *Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Minneapolis, MN, 2009*, pp. 5203-5206.
- [41] T. D. Tan and N. Van Tinh, "Reliable Fall Detection System Using An 3-DOF Accelerometer and Cascade Posture Recognitions," *Signal and Information Processing Association Annual Summit and Conference (APSIPA), Asia-Pacific, Siem Reap, 2014*, pp. 1-6.
- [42] N. Rathi, M. Kakani, M. El-Sharkawy and M. Rizkalla, "Wearable Low Power Pre-Fall Detection System with IoT and Bluetooth Capabilities," *IEEE National Aerospace and Electronics Conference (NAECON), Dayton, OH, 2017*, pp. 241-244.
- [43] Y. Zigel, D. Litvak and I. Gannot*, "A Method for Automatic Fall Detection of Elderly People Using Floor Vibrations and Sound—Proof of Concept on Human Mimicking Doll Falls," *IEEE Transactions on Biomedical Engineering*, 2009, Vol. 56, No. 12, pp. 2858-2867.
- [44] H. Rimminen, J. Lindström, M. Linnavuo and R. Sepponen, "Detection of Falls Among the Elderly by A Floor Sensor Using the Electric Near Field," *IEEE Transactions on Information Technology in Biomedicine*, 2010, Vol. 14, No. 6, pp. 1475-1476.
- [45] Y. Li, K. C. Ho and M. Popescu, "Efficient Source Separation Algorithms for Acoustic Fall Detection Using a Microsoft Kinect," *IEEE Transactions on Biomedical Engineering*, 2014, Vol. 61, No. 3, pp. 745-755.

- [46] J. Clemente, F. Li, M. Valero and W. Song, "Smart Seismic Sensing for Indoor Fall Detection, Location, and Notification," *IEEE Journal of Biomedical and Health Informatics*, 2020, Vol. 24, No. 2, pp. 524-532.
- [47] Y. Ogawa and K. Naito, "Fall Detection Scheme Based on Temperature Distribution with IR Array Sensor," *IEEE International Conference on Consumer Electronics (ICCE)*, Las Vegas, NV, USA, 2020, pp. 1-5.
- [48] C. Lin, S. Wang, J. Hong, L. Kang and C. Huang, "Vision-Based Fall Detection through Shape Features," *IEEE Second International Conference on Multimedia Big Data (BigMM)*, Taipei, 2016, pp. 237-240.
- [49] S. C. Agrawal, R. K. Tripathi and A. S. Jalal, "Human-Fall Detection from An Indoor Video Surveillance," *2017 8th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, Delhi, 2017, pp. 1-5.
- [50] A. Lotfi, S. Albawendi, H. Powell, K. Appiah and C. Langensiepen, "Supporting Independent Living for Older Adults; Employing A Visual Based Fall Detection Through Analysing the Motion and Shape of the Human Body," *IEEE Access*, 2018, Vol. 6, pp. 70272-70282.
- [51] K. Gunadi, Liliana and J. Tjitrokusmo, "Fall Detection Application Using Kinect," *International Conference on Soft Computing, Intelligent System and Information Technology (ICSIIIT)*, Denpasar, 2017, pp. 279-282.
- [52] T. Tsai and C. Hsu, "Implementation of Fall Detection System Based on 3D Skeleton for Deep Learning Technique," *IEEE Access*, 2019, Vol. 7, pp. 153049-153059.