



生醫量測系統設計

基於輕量化CNN之嵌入式心律異常量測系統

輕量化，可於嵌入式系統中量測的系統

與會者

課堂學生、教授

報告者

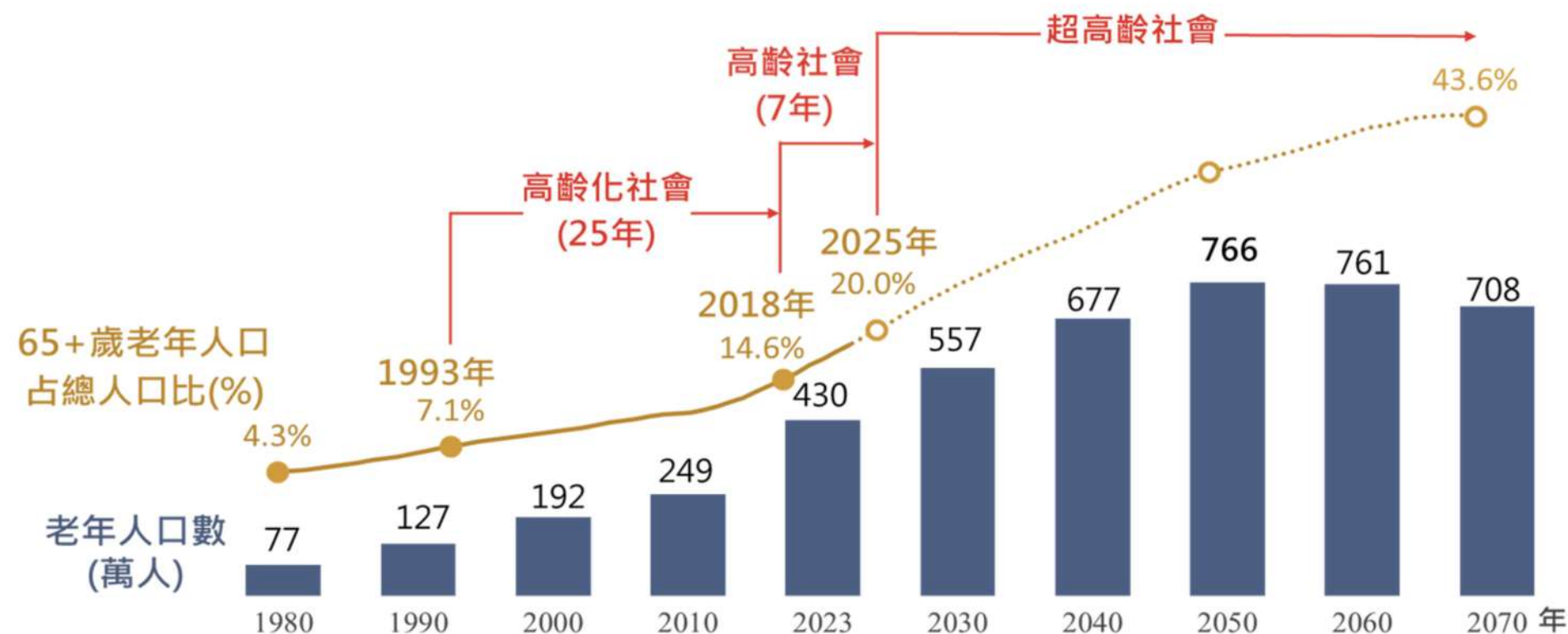
張仁傑、林家慶、吳峻杰



目錄

- | | |
|------------------|-------------------|
| 01 專題動機 | 04 硬體實現方法 |
| 02 系統架構 | 05 實驗結果 |
| 03 <u>軟體實現方法</u> | 06 <u>未來展望與討論</u> |

專題動機



註：截至2023年為實際值，其餘為假設總生育率為1.2人之中推估結果。

圖片來源:國家發展委員會統計

- 根據台北榮總的報告指出，台灣有23萬人罹患心律不整
- 根據台灣醫學會報告指出，心房顫動是老年人盛行率最高的心律不整

專題動機： 現行方案



Xiaomi 手環 9 Pro

GNSS定位 泳姿辨識

NT\$1,799

立即購買

瞭解更多 >

圖片來源:小米官方網站

現行市場上相對平價的解方，可以偵測心率
但卻無法辨認是否有心律不整或其他病徵



Venu 3

GPS 智慧腕錶

產品料號 010-02784-20

NT\$ 14,990



活力白 - 45mm

小錶徑版

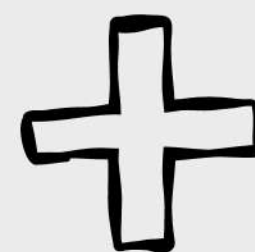
圖片來源:Garmin官方網站

現行市場上的解方，可以達成偵測心率和辨認，但也包含了許多功能而價格不斐

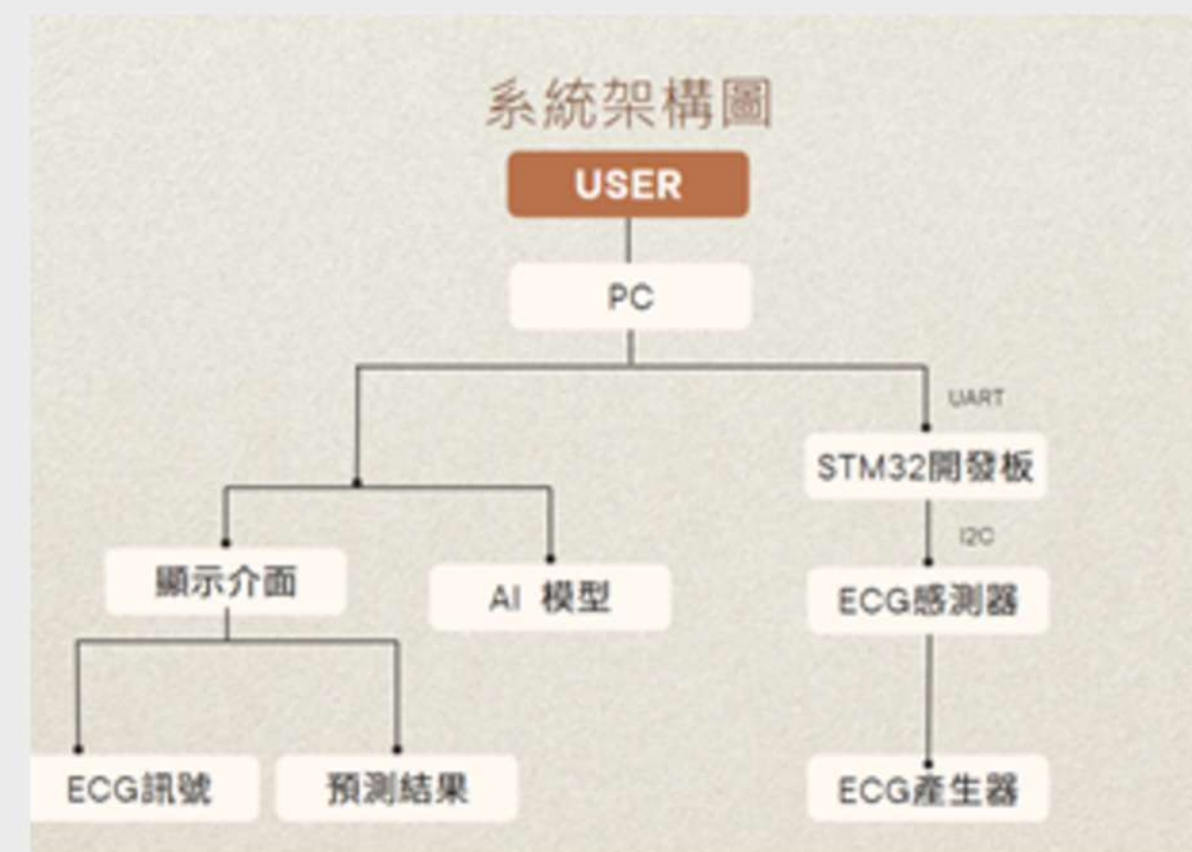
我們的想法

簡單且便宜的 solution

利用MCU + AI 製作成行動穿戴裝置來方便長者進行居家檢測，系統可以偵測不同的心律不整病徵，也可以減少醫療資源浪費

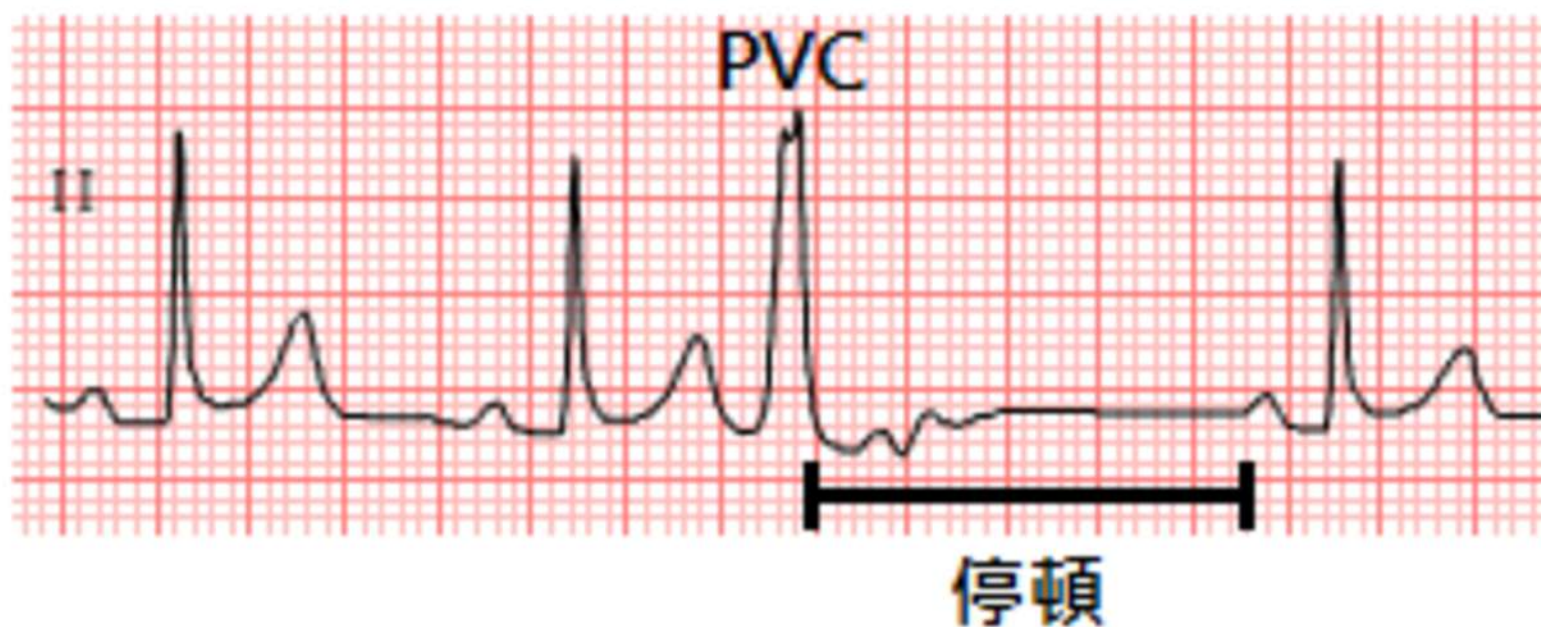


CNN with
PyTorch



實驗的系統架構圖，由感測器接收訊號並以I2C協定傳輸至開發板並進行訊號處理，接著透過UART協定輸入至電腦中由預訓練並經過壓縮的模型進行訓練

本次實驗主要判斷病徵



心室早期收縮(PVC) 圖片來源:陳煌奇診所網站

心室的異位節律點發出激動波使心室提早收縮，使QRS綜合波在P波前出現。此種收縮也會產生脈搏。且此種傳導較慢，在心電圖上會看見較寬的QRS波（寬大約2至4倍）。



心房早期收縮（PAC）圖片來源: LITFL網站

P波提前出現，形狀異常或倒置，QRS波行為正常但RR peak會小於正常心跳間隔

Mitbih資料庫讀取

介紹:

MIT-BIH 心律失常資料庫 (Arrhythmia Database) 包含 48 個半小時長度的心電圖紀錄，每個紀錄皆由 WFDB 標註檔 (.atr) 給出所有 R 峰 (心跳) 與其對應的心律類型標記。整個資料庫加總起來，總共有約 109,488 個標註點 (beat annotations)。

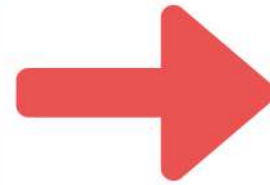
方法

1. 迴圈讀取每筆 Record ID (例如 100 ~ 234 總共 48 筆標示檔案)。
2. 每筆 Record 用 wfdb.rdrecord、wfdb.rdann 取出訊號與標註。
3. 對每個 R -peak做切片 (187 點) 並轉成 numpy 陣列。
4. 以對應符號做映射得到整數標籤。
5. 把所有切片與標籤累積到共同的 list。
6. 轉成 DataFrame 並 to_csv，並以80%，20%的方式分成訓練集和資料集

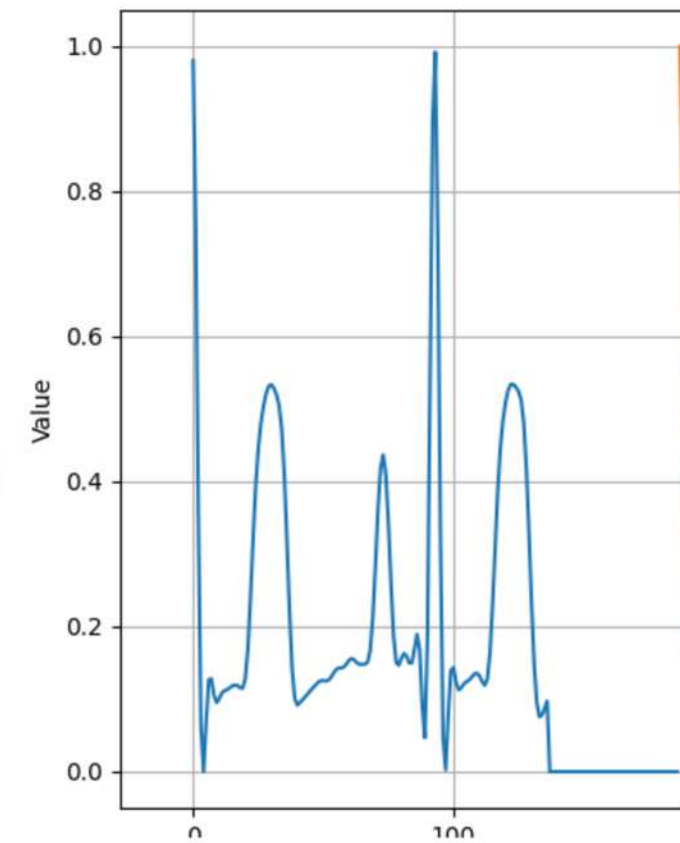
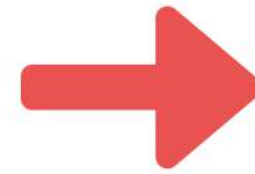
軟體實現(CNN方面)



Database
其中標記分為
N(Normal)
PAC(房性早搏)
PVC(室性早搏)
Fusion beat (融合搏動)
Unknown beat (未知類型搏動)



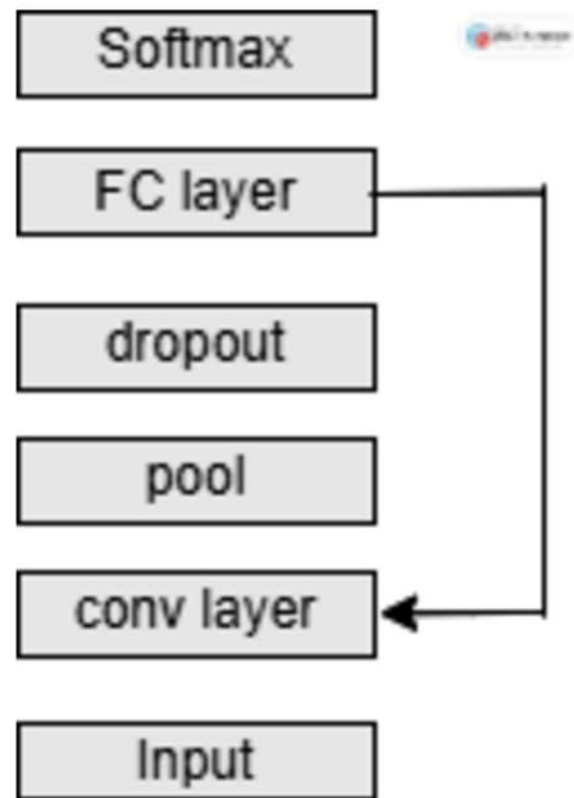
降採樣至125 Hz，因為一般ECG訊號約為0.05~40Hz之間，這個取樣頻率以符合奈奎斯定理，這個取樣頻率可以良好的取得心律特徵並節省運算量。



數據正規化以加速收斂和防止模型對特定特徵產生過擬合的現象

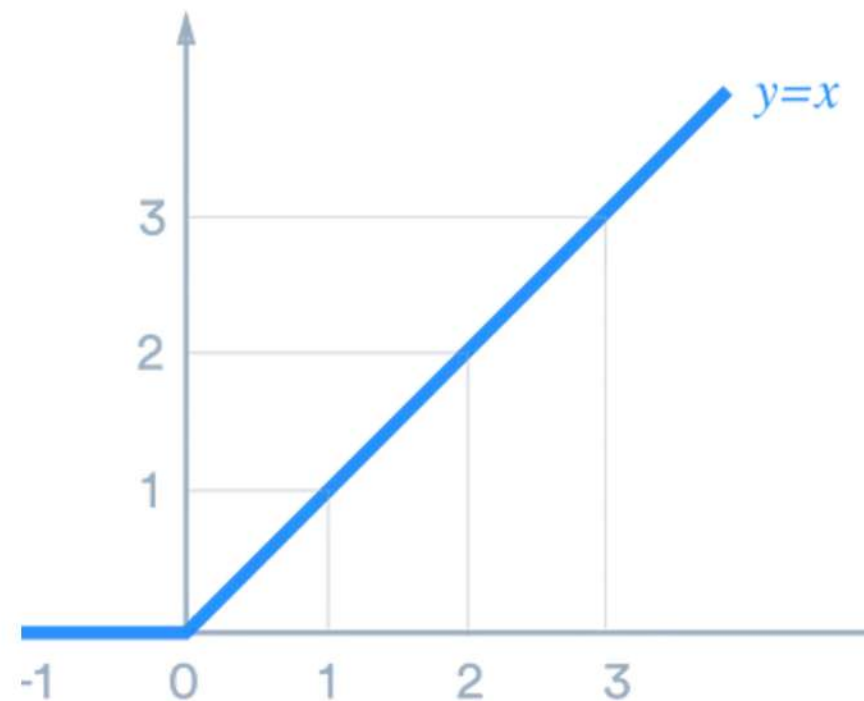


軟體實現



實作CNN 架構圖，其中參考VGG Net架構透過逐層特徵提取和小卷機核特徵堆疊可以達到更好的訓練效果。

EX: kernel = 3, conv=32



我們以ReLU 做為 Activation function，sigmoid 和tanh相較於ReLU容易飽和，所以選擇了此方案



Adam 根據隨機梯度下降算法(SGD)的集成，結合了RMSProp, 所選擇了Adam 做為計算學習率的工具。



軟體實現(QAT方面)

量化感知訓練（QAT）是一種在訓練階段就模擬量化誤差的技巧，讓模型在訓練過程中學習如何適應整數化後的行為，這樣在部署成 INT8 模型後仍能維持原本精度。

為什麼要量化模型

1. 降低模型大小

原始模型使用 32-bit 浮點數（FP32）來儲存權重與中間值。

透過量化，可以將其轉換為 8-bit 整數（INT8）甚至更低（如 INT4）。

這樣能直接將模型大小減少 4 倍（FP32 → INT8），對於儲存空間有限的設備非常有幫助。

2. 提升推理速度

整數運算在大多數硬體（如 ARM Cortex-M、DSP、NPU、TPU）中執行速度比浮點數快得多。

這可帶來 2~4 倍以上的推論加速，特別適合於手機、IoT 裝置、微控制器等。

3. 減少能源消耗

整數運算比浮點數耗電更少。

在電池供電或對電力敏感的裝置（如穿戴式裝置、醫療設備）中，量化可以大幅降低功耗，延長裝置壽命。

缺點

因為原模型是浮點數精度，所以會損失一些精度

QAT(訓練中量化)

方法

一邊訓練一邊量化，所以可以將誤差的量化結果投入下一次的訓練

優點

精度損失低

相對穩定，不依賴 Calibration Dataset

缺點

需要重新訓練模型，訓練時間成本高

PTQ(訓練後量化)

方法

模型訓練好後再量化，因此訓練速度快，但因為是事後才量化不會考慮到量化誤差，所以結果模型誤差較大

優點

不必重新訓練，時間成本低

缺點



精度損失高

量化後結果

```
Epoch 030 | Train Loss: 0.0545 | Val Acc: 0.9837 | Val F1: 0.9171  
Save best model.
```

```
Epoch 45 | Train Loss: 0.0863 | Val Acc: 0.9808 | Val F1: 0.8845  
saved best_qat_model.pth
```

上面是未經量化的精準度
下面是經過量化的精準度

 mitbih_cnn_int8.pt	2025/5/29 下午 08:53	PT 檔案	299 KB
 mitbih_cnn.pth	2025/5/29 下午 08:53	PTH 檔案	1,043 KB

mitbih_cnn_int8: 經過QAT量化過的INT8整數模型

mitbih_cnn: 未經量化的FP32浮點數模型

差距約四倍: FP32 → INT8

降低模型大小可以讓我們成功將模型放入嵌入式裝置(stm32f401rct6)開發版中

量化相關文獻

1. Quantization and Training of Neural Networks for Efficient Integer-Arithmetic-Only Inference

提出 TensorFlow Lite 的整數量化技術。

比較 PTQ 與 QAT，發現 QAT 能顯著減少量化導致的準確率下降。

MobileNet 的 Top-1 accuracy 在 PTQ 時會掉 3~5%，但經過 QAT 則下降不到 1%

2. BRECQ: Pushing the Limit of Post-Training Quantization by Block Reconstruction

主題是 PTQ 優化，但同樣提到 QAT 在極低位元（如 INT4）下仍明顯優於 PTQ。

實驗中，ResNet-18 使用 QAT 可以保留 70% 以上準確率，而 PTQ 僅約 60%。

3. HAWQ-V3: Dyadic Neural Network Quantization

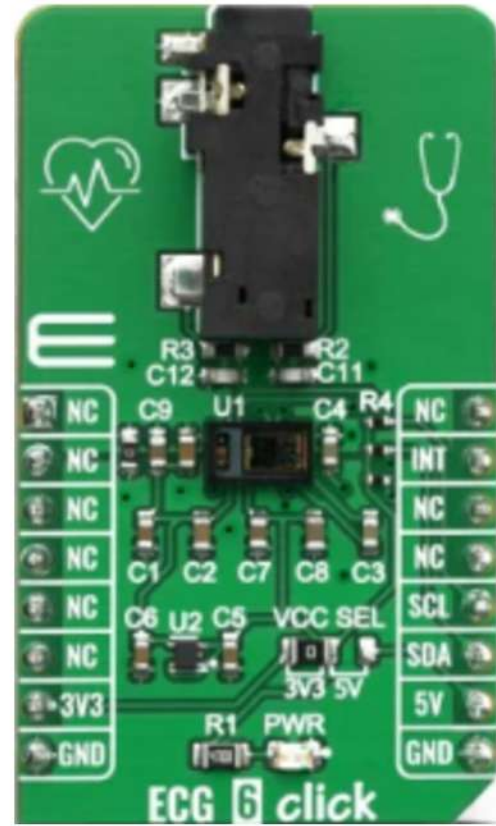
明確指出 QAT 通常比 PTQ 多出 2~5% 的 top-1 accuracy。

提供量化後模型大小（例如 32-bit → 8-bit 模型大小減少 75%）

硬體實現



ECG訊號藉由
ECG產生器產生
各種病徵的訊號



ECG訊號將會由
Max86150 感測
器蒐集



使用PyTorch訓練模型產生
ONNX模型檔



Cube AI可以協助將模型放入開發版

遇到的困難

開發版記憶體為 64K SRAM + 256K FLASH

模型大小為286K，理論上是放不進板子的FLASH，但是STM32CUBE.AI還是成功將模型放入板子，但是結果無法正常執行，不確定是SRAM太小導致運算空間不足還是FLASH太小導致模型無法正確放入導致結果無法正常運行。

解決方法->將模型在電腦上執行，或是可以試試縮減捲積或INT4方法

Tensorflow對每個函式庫版本要求極高

像是CUDA、CUDNN、Python、Keras、tensorflow_model_optimization等等，只要有一個版本出問題就無法正常執行模型訓練，前置作業過於複雜且不易除錯

解決方法->使用PyTorch進行模型訓練

模型驗證

此圖為根據訓練出來的模型放入test data後所產生的 confusion matrix，其中標記N為正常、S為PVC、V為PAC、F為Fusion、Q為unknown，row為真實資料樣本數，column為模型判斷的資料筆數，對角線為模型判斷正確數量

N資料樣本數為: 4617筆 44%，模型判斷正確數為: 4601 99.7%

S資料樣本數為: 1953筆 19%，模型判斷正確數為: 1415 72.4%

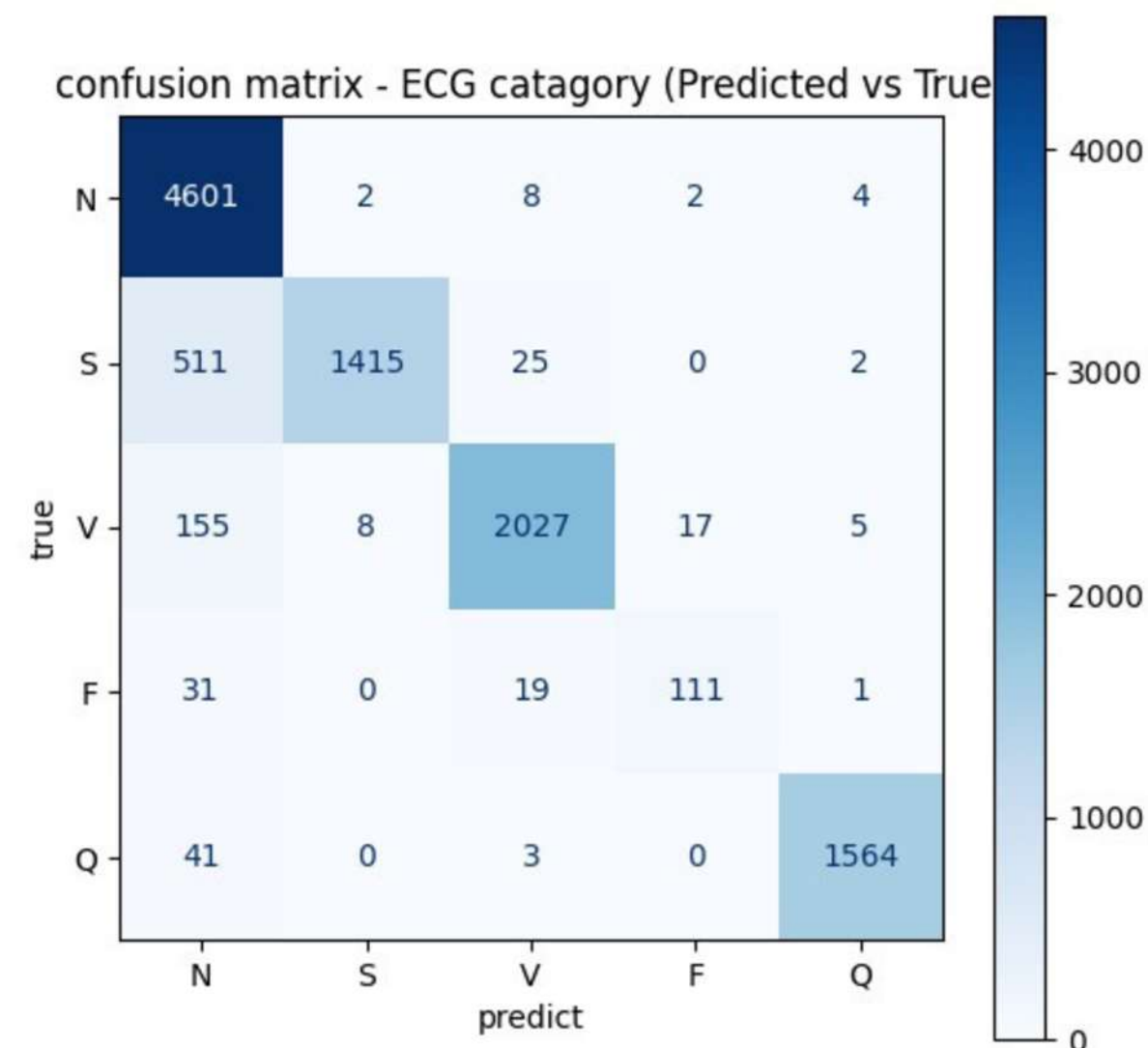
V資料樣本數為: 2212筆 21%，模型判斷正確數為: 2027 91.6%

F資料樣本數為: 162筆 1.5%，模型判斷正確數為: 111 68.5%

Q資料樣本數為: 1608筆 14.5%，模型判斷正確數為: 1564 97.3%

正常樣本與病徵樣本比率約為正常4617(44%)、帶病徵5935(56%)

正確率為94.57%

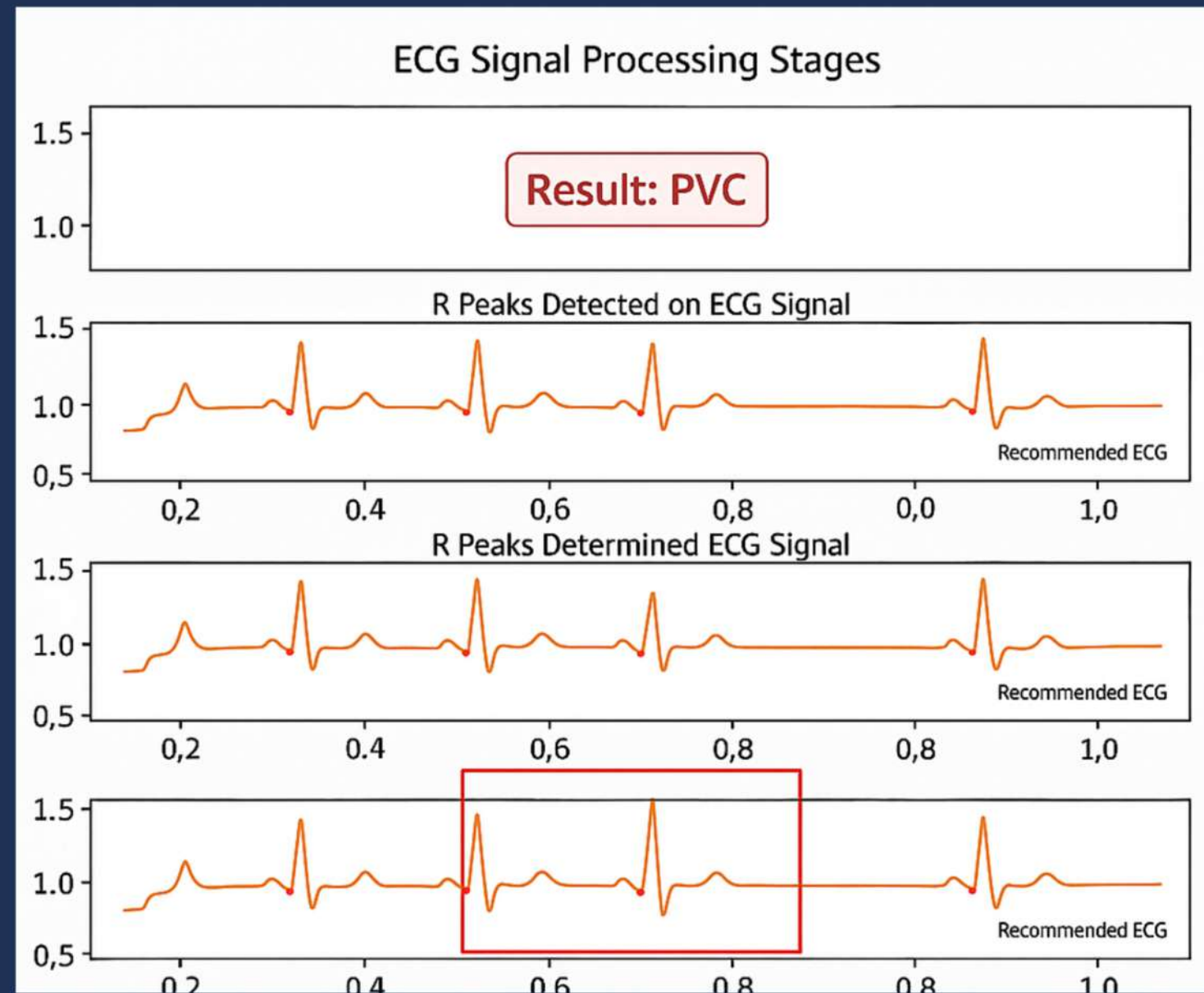


實驗方法

- 我們隨機抽取原始訓練資料的20%來進行標記的驗證
- 在外部輸入的部分，我們藉由ECG產生器分別產生60~120Hz正常心跳各200筆，以及PAC,PVC的異常心跳也各200筆，並且利用R-peak尋找演算法定位，以峰值為起點做紀錄，做為測資
- 最後將資料打亂並隨機輸入模型得到以下結果

實驗結果

最終QAT onnx檔案, 模型大小286KB的版本正確率為 **94.57%**



結果將以波型方式呈現，框出有病徵的片段，並且告訴其病徵為何

分工表

張仁傑

簡報製作、模型訓練、韌體開發

林家慶

簡報製作、模型訓練、韌體開發

吳峻杰

簡報製作、模型訓練、韌體開發

課堂專題報告構思

基於輕量化CNN之嵌入式
心律異常量測系統

THANKS FOR WATCHING