# 基於 ShuffleNet 之道路事件聲音識別輕量化邊緣智慧系統

Shih-Ting Tai
Dept. of Electrical Engineering
National Central University
Taoyuan, Taiwan
taishihting@g.ncu.edu.tw

Yu-Cheng Wu
Dept. of Electrical Engineering
National Central University
Taoyuan, Taiwan
mineeric@g.ncu.edu.tw

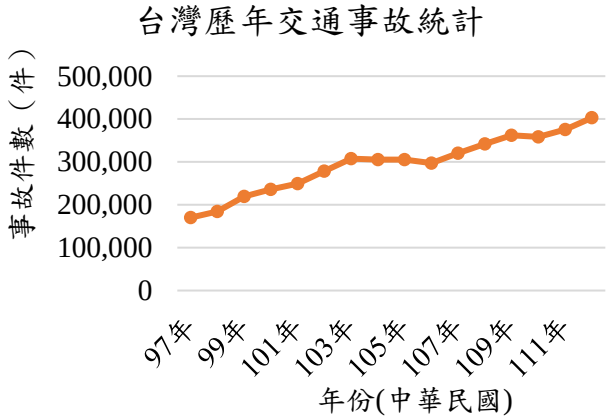
Kai-Fang Cheng
Dept. of Electrical Engineering
National Central University
Taoyuan, Taiwan
kevin71xb37@g.ncu.edu.tw

Yi-Chuan Lu
Dept. of Electrical Engineering
National Central University
Taoyuan, Taiwan
yichuan921030@g.ncu.edu.tw

摘要—在現代社會中，隨著交通流量急速增加，道路事故發生頻率也相應上升。然而，現有監視系統存在視覺死角，後端影像辨識難以感知監視器鏡頭以外的事故發生，這對即時識別和救援效率提出了挑戰。為解決此問題，我們提出—基於 ShuffleNet 之道路事件聲音識別輕量化邊緣智慧系統，利用聲音傳播特性補足視野死角問題，而形成交通事故監測之完整解決方案。

## I. 前言

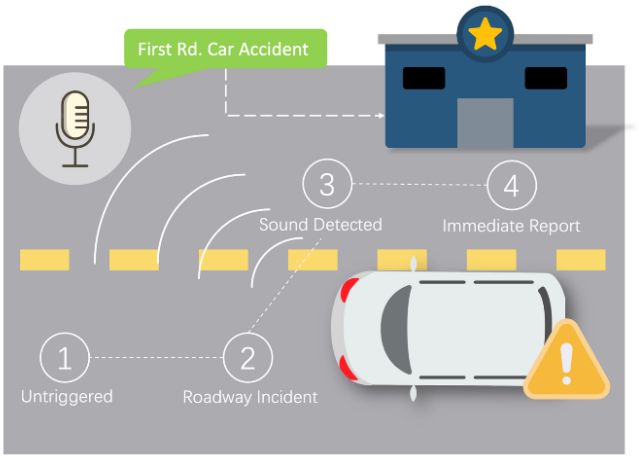
根據交通部統計，全國交通事故總件數從民國 100 年的 235,776 件急遽增長至 111 年的 375,844 件，因交通事故受傷的人數也平均每年增加約 16,834 人，趨勢如圖(一)。面對日益增長的交通事故，監控系統能否即時發現、通報並展開救援成為重要的道路安全議題。因此監控系統若能具有準確的事故偵測與自動通報能力，將有望提升救援團隊應對事故的效率。




圖(一) 台灣歷年交通事故統計

目前的道路監控系統已可通過在遠端監控中心進行影像辨識，能夠有效辨識車禍事故的發生並迅速啟動後續的救援行動。這種方式縮減了對人工通報的依賴，提高了交通事故後警察和救護人員等相關機構的反應速度。然而，傳統的監視器卻存在「視覺死角」的限制，這意味著在事故發生的特定位置，監視器可能無法拍攝到完整的情況。因為後端的影像辨識無法感知監視機鏡頭以外的事故發生，很多交通事故仍然需要依賴目擊者的通報，致使傷患無法及時得到救助。

因此本專題提出道路事件聲音識別系統，充分利用聲音傳播「無死角」的特性，可以偵測到以往無法觸及的死角，成品概念圖如圖(二)。另外，我們將在監視器端進行邊緣運算，即時偵測事件的發生，避免等待訊號回傳到監控中心經過處理再進行辨識的過程。這樣可以節省寶貴的反應時間，有助於加速警察和醫療的救援行動，提高對事故傷患的即時救助效果。




圖(二) 專題成品概念圖

具體而言，本系統融合軟硬體整合開發。軟體端進行高準確率之 ShuffleNet 神經網路參數訓練，其中包含資料集的建立與預處理，以及在追求高準確率的同時，還須兼顧神經網絡的運算複雜度，以滿足邊緣運算需求。硬體端則匯入訓練好的參數並建立高度輕量化的 ShuffleNet神經網路模型，從而改善當前高成本的聲音辨識硬體架構。因此，本專題之目標是在資源有限的 FPGA 板上追求高效且輕量化的神經網路聲音辨識模型，以符合實際需求。

## II. 文獻回顧

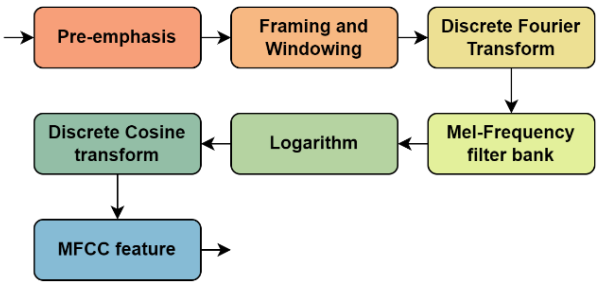
本章節將會對我們專題中所用之(A)聲音預處理技術(MFSC)、(B)深度學習神經網絡、(C)ShuffleNet 神經網絡以及(D)聲音辨識相關的文獻文獻回顧。

A. 聲音預處理技術

聲音訊號主要可分為時域與頻域兩種。在頻域分析中，訊號被分解成一個或多個頻率分量，每個分量的功率參數都會被詳細展示，這使得分析者能夠明確了解訊號中各個頻率成分的特性。相對於頻域，時域分析則關注訊號在時間軸上的整體變化，將所有頻率成分綜合在一起，以觀察訊號參數如何隨時間變化。而在這些特徵提取技術可以被分為兩類[1]，第一種是依「語音產生方式」，例如: 線性預估編碼(LPC) [2]；第二種是依「語音感知」，例如: 取梅爾頻率倒譜係數(MFCC)與本專題所使用之 MFSC。

- MFCC 介紹

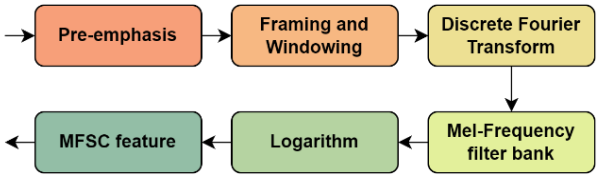
取梅爾頻率倒譜係數(MFCC)流程如圖(三)，預強化(Pre-emphasis)補償訊號高頻損失，是整個音頻處理中非常重要的首要步驟。接下來，將訊號拆分成幀(Framing)，這是為了實現對訊號的穩定分析；隨後進行加窗(Windowing)操作，可依需求選擇漢寧窗或漢明窗，並搭配適當窗口大小和重疊時間，這有助於增強諧波、平滑邊緣並減少邊緣效應。透過離散傅立葉變換(DFT)，將訊號由時域轉為頻域，並計算訊號的頻譜功率分佈。Mel 帶通濾波器是一組基於音高知覺的濾波器，類似人耳對語音的感知，旨在提取語音訊號的非線性表示，且慣例的 Mel 濾波器組由 40 個三角形濾波器構成[3]。最後，取對數後使用離散餘弦變換(DCT)，以選擇梯度變化最大的係數，或深入分析對數頻譜與濾波器組之間的關係，提升語音識別的精確度。




圖(三) MFCC 流程圖

- MFSC 介紹

與 MFCC 不同之處在於，MFSC 省略了離散餘弦變換(DCT)步驟，直接採用音頻流的倒譜表示作為語音特徵，流程圖如圖(四)。這不僅簡化計算過程，同時也保留了更多原始語音數據[2]，使得頻譜能量可以更好地保持局部特性。儘管如此，MFSC 直接使用對數能量的方法仍然為語音特徵提取提供了一種有效的方案，在特定應用中可能更適用於保留訊號的本地特性。基於其對 MFCC 的運算簡化及特徵存取特性，因此我們採用 MFSC 作為本系統之聲音預處理工具。

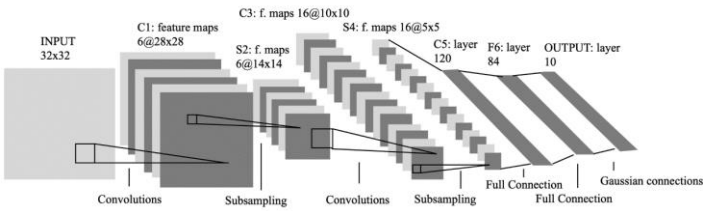



圖(四) MFSC 流程圖

B. 深度學習神經網路

神經網路（Neural Network, NN）是一種模仿生物神經系統的數學模型，由眾多運算節點（稱為神經元）構成，並通過可調節的權重連接。這些神經元按層次組織，包括輸入層、一個或多個隱藏層和輸出層。當有多個隱藏層時，這種結構被稱為深度學習（Deep Learning）。神經網路已廣泛應用於各種傳統程式難以處理的領域，如視覺和語音辨識，因其能夠學習並調整內部權重以適應外界資訊，從而提高處理這些複雜問題的效率和準確度。

深度學習[5]中常見的兩種網路為遞迴神經網路（RNN）和卷積神經網路（CNN）。RNN 主要用於語音和自然語言處理[6]，具有記憶先前輸入的能力，適合處理時間序列數據。然而，其容易產生梯度消失問題，可通過長短期記憶（LSTM）技術[7]解決。另一方面，CNN 廣泛應用於視覺處理[8]、語音處理[9]和物件識別等，其強大之處在於能自動從數據中提取特徵，結構包括卷積層、池化層和全連接層，如 LeNet 神經網路架構[10]。

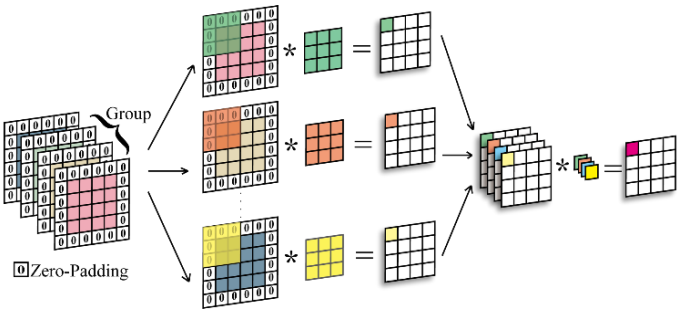



圖（五）LeNet [10]卷積網路模型架構圖

C. ShuffleNet 神經網絡

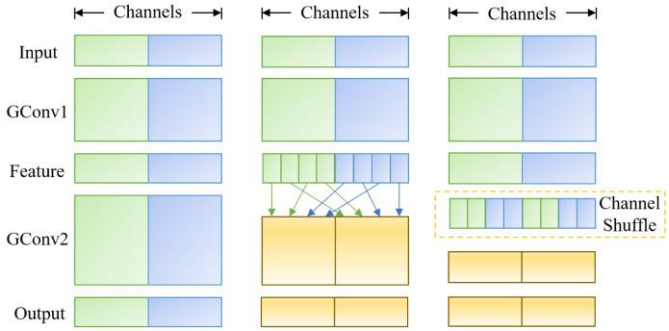
在硬體的實時邊緣運算下，神經網絡的運算特性與參數數量多寡尤為重要，因此[11]提出的 MobileNets，此神經網絡不僅擁有較輕便的模型參數數量，也兼顧較高的運算速度，以

追求能在行動裝置(Mobile)上實行，符合本專題之邊緣運算需求。其核心概念是將傳統卷積的提取特徵過程拆成 Pointwise Convolution 和 Depthwise Convolution 兩步驟進行，合稱為深度可分離卷積，這種拆分使得模型在保持性能的同時能大幅降低計算複雜度和模型大小，適用於資源有限的設備，運算過程如圖(六)所示。




圖(六) Pointwise & Depthwise Convolution

ShuffleNet [12]是基於 MobileNets 的概念發展的高效神經網路架構，特別是針對邊緣裝置的需求設計。它結合了群組卷積 Group Convolution 和通道洗牌 Channel Shuffle 兩種技術來達到高效的模型輕量化，同時保持良好的模型表現，其概念實現方式如下圖(七)。




圖(七) Group Convolution & Channel Shuffle

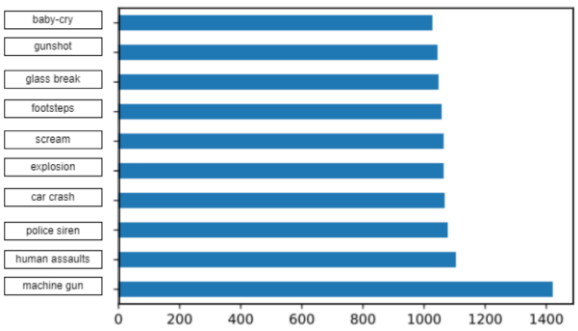
群組卷積將傳統卷積拆分成兩個部分，首先對各個通道進行單通道的獨立二維卷積，取得區域的特徵；接著，使用 1×1 的卷積核對這些輸出進行卷積運算，強化特徵之間的關聯。最後，通過通道洗牌促使資料交換，從而使模型可以在大幅減少運算量和參數儲存量的前提下學習更複雜的特徵。

[13]在 FPGA 上成功開發 ShuffleNet 神經網絡 ，利用咳嗽聲音特徵進行疾病種類的辨識，最終可在 7.9K 個 LUT、12.9K 個 Flip-Flop 和 41 個 DSP 達到了 86.5% 的精確度，顯示出 ShuffleNet 在硬體端的潛力。

D. 聲音辨識文獻回顧

聲音作為自然界中常見的訊號之一，蘊含豐富的資訊，可協助人們發現潛在危險並適時做出反應，因此聲音識別技術被廣泛應用於各領域。其中語音視覺化[14]運用自然語言處理(NLP)將語音即時轉換成文字，並提高 67%正確率，幫助吶語症患者正常生活。聲音事件追蹤和定位[15]運用 T 型麥克風陣列在嘈雜環境中，仍有 93%精準率可定位聲音發生來源。另有，融合聽覺與視覺的事件辨識[16]，藉由 Audio-Video Concurrence (AVC) 與 K-nearest neighbors (KNN)應用，使影像結合聲音再進行辨識，讓深度學習模型判定出更準確且更有意義的結果。

在道路安全方面，事件的影像辨識可能會有死角或是視線不佳等問題而無法捕捉到周遭的車禍事件，因此城市道路事件的聲音辨識亦是一重要議題。然而，道路大多時間相當嘈雜，所以事件的聲音樣本中常帶有大量雜訊，這會使辨識的準確度大為降低，故[17]使用 OneClass SVM 分離事件聲音與雜訊，並藉由 DNN 可辨識出是否有危險事件發生。[18]提出 Unusual Occurrences in Audio Forensics Database (UOAFDB)資料集，其組成聲音分布如圖(八)，並為各種特殊事件聲音加上 15 種背景環境音效，並運用 MFCC 與 CNN，辨識正確率可達到 81.50%。不同於上述兩篇論文僅為軟體層面實作，[19]致力於聲音辨識的邊緣化運算，優化 MFSC 與 CNN 硬體設計，成功減少 65.63%的 CNN 參數資料儲存空間，實現可在 FPGA 運行的輕量化運算。




圖(八) UOAFDB 道路聲音資料集[18]

在道路事件中，車禍往往具有嚴重性，需要即時的救援和處理。車禍的聲音通常分為兩類：一類是持續性聲音，如輪胎與路面的尖銳摩擦聲；另一類則是短暫突發的聲音，如撞擊

聲。為了有效辨識這些聲音，[20]提出雙層聽覺辨識，運用 bag-of-words 方式辨識兩類聲音，此方法考慮到了車禍聲音的多樣性，並成功區分了不同特徵的聲音模式，可達 78.95%正確率。Crashzam[21]使用車內錄製的車禍聲音作為資料集，並且該系統具有高度可攜帶性，能夠在智慧型手機上運行。這種方便的部署方式提供了更靈活的應用場景，使車禍聲音辨識系統更具實用性。ENCAP (European New Car Assessment Programme)資料集被[22]區分為四種車禍類型，並只用了 20 個 MFCC 參數，在 SVM 演算法下得到 67.22%的正確率。這表明在考慮不同車禍情境時，特定的特徵和演算法的選擇仍是個值得深入研究的議題。

## III. 聲音資料集

由於車禍事件的發生通常是突發性且持續時間極短，再加上人為近距離拍攝或錄音存在顯著的安全風險，因此車禍聲音的資料集相當稀少且難以取得。為了克服這一困難，本專題選擇利用真實的聲音資源，如 ESC-50[23]中所提供之城市噪音與 Car Crashes Time[24]中所擷取特定車禍音檔；搭配人造音效如 Soundsnap 中的 Car Crash Sound Effects[25]，作為車禍的聲音資料集。

### A. *ESC-50*

ESC-50 是一個用於環境音分類的標註資料集，包含 2000 條音頻錄音，每條長度為 5 秒。共 5 個主要類別：動物、自然音景和水聲、人類非語音聲音、家居聲音、城市噪音；本專題可取其「城市噪音」類別作為資料集中非車禍事件的部分。

### B. *Car Crashes Time*

Car Crashes Time 是一個致力於推廣道路安全的 YouTube 頻道，其中包括了至少六年的各種車禍合集影片，每部影片 10 分鐘，共 28 部影片，提供大量車禍錄像，且部分片段具車禍事件聲音；由於大部分影片皆為車內行車紀錄器所拍攝，所以聲音效果並不清晰，仍需經擷取與篩選才可使用。

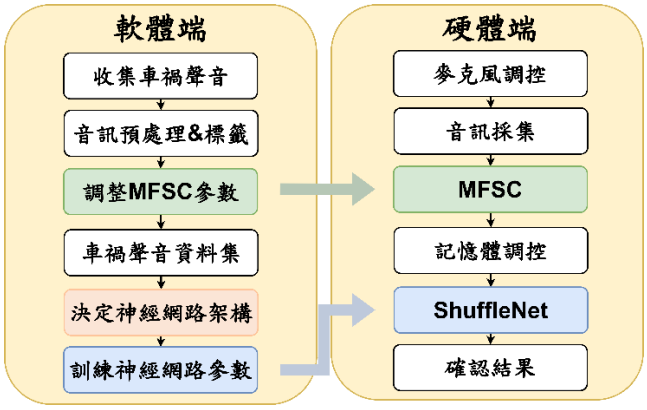
### C. *Soundsnap Car Crash Sound Effects*

Soundsnap 是一個專業的音效和音樂資源分享網站，從 2008 年開始營運至今，提供種類豐富的音樂素材。其中超過 100 個汽車碰撞聲音，包括簡單的撞擊和更複雜的情境，如輪胎尖銳摩擦聲、打滑聲、喇叭鳴笛聲；特色為聲音特徵明確，無背景雜音干擾。

由於上述之資料集仍不夠充足，可能導致深度學習模型訓練效果不佳，產生 overfitting 問題，因此我們採取 Data augmentation 的手法來擴充與平衡資料集[26]，其中可採取的手法如:對時間(Time)進行等比縮放(Stretch)、對時間與聲音強度平移(Shift)、或者加入背景噪音(Noise)，共可分為 TimeStretch, TimeShift, PitchShift, AddBackgroundNoise 四種[27]。

## IV. 實驗方法

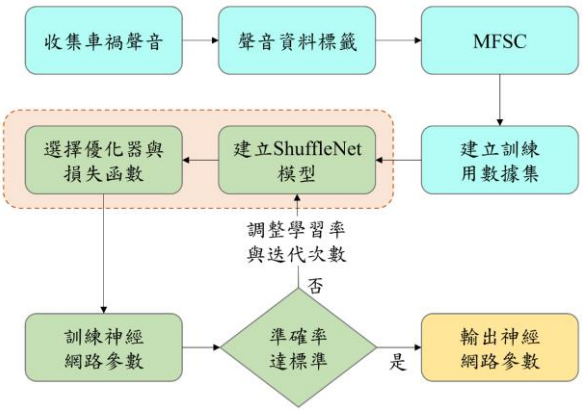
本系統結合軟硬體開發，架構如圖(九)。軟體部分涵蓋建立及預處理資料集並訓練出高辨識準確度的 ShuffleNet 神經網路，同時還需考慮運算複雜度以滿足邊緣運算的需求。硬體部分則將訓練好的模型參數整合至輕量化的 ShuffleNet 架構中，提升聲音辨識系統效能。




圖(九) 整體系統架構圖

### A. 軟體端

欲將聲紋辨識的卷積神經網路實現於硬體，需在軟體端預先訓練卷積網路的所有參數，再將參數匯入硬體端的記憶體中。現今聲紋辨識的主要辨識資料為音訊的頻譜，然而我們通常只能直接獲得一段時間區間中的聲音訊號，因此我們需要將時域的聲音訊號轉為頻域的頻譜分布，這個步驟稱為「聲音預處理」。現今以 Mel-frequency spectral coefficient (以下稱 MFSC) 聲音預處理流程最廣為使用，因此本系統透過 MFSC 將音訊資料轉換為頻譜，再將頻譜作為音訊資料集進行軟體端的卷積神經網路模型的訓練與測試，軟體端設計圖如圖(十)。




圖(十) 軟體端設計流程圖

- MFSC

為了提升音訊資料集對模型訓練的成效，我們會先對資料集做 MFSC，此處的關鍵為梅爾過濾器，此濾波器是根據人耳聽覺特性設計，為非線性濾波，對於不同頻率的聲音敏感度不同。因此，梅爾濾波器是由多個不同中心頻率的脈波重疊而成，並非像大多數過濾器由覆蓋所有頻率的單一波形構成，有助於去除特定頻率區域內的資料，從而獲得更具代表性的頻譜特徵。梅爾濾波器組的公式如下所示，m 為濾波器編號，$f_m$ 為第 m 個濾波器的中心頻率。從下方公式可看出，在第 m 個濾波器中，若頻率 k 不在頻率 $f_{m-1}$ 及 $f_{m+1}$ 之間，則會完全過濾該頻率的振幅。反之，若頻率 k 介於頻率 $f_{m-1}$ 及 $f_m$ 之間或介於 $f_m$ 及 $f_{m+1}$ 之間，則根據公式計算該頻率組成振幅的放大倍率，並且，從公式中可看出中心頻率的放大倍率為最大。

$$B_m[k] = \begin{cases} 0 & \text{for } k < f_{m-1} \text{ and } k > f_{m+1} \\ \dfrac{k - f_{m-1}}{f_m - f_{m-1}} & \text{for } f_{m-1} \le k \le f_m \\ \dfrac{f_{m+1} - k}{f_{m+1} - f_m} & \text{for } f_m \le k \le f_{m+1} \end{cases} \quad (1)$$

- ShuffleNet 卷積神經網路

由於 MFSC 後的頻譜資料為一維矩陣，不利於神經網路模型的卷積及池化。因此，本系統將每 64 幀頻譜水平堆疊，形成方形頻譜圖，有助於神經網路運作。再將這些頻譜圖作為資料集，分為訓練集以及測試集，避免測試資料集遭到汙染，影響測試結果的可信度，再將這些資料放入卷積神經網路訓練及測試。

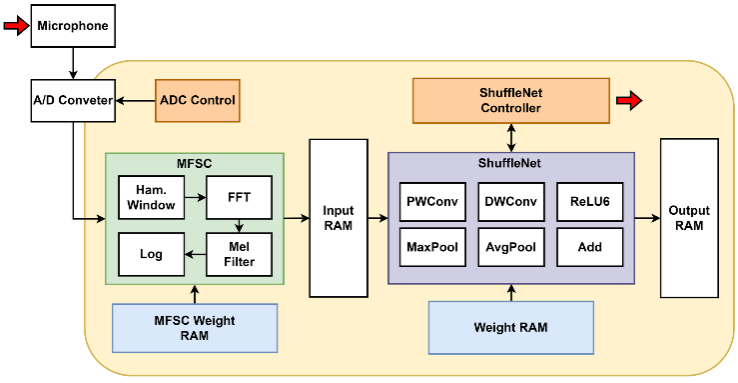
由於並非所有的神經網路架構都是用於辨識音訊，我們使用 ShuffleNet 卷積神經網路，其適用於較短音訊資料的辨識，並且其參數量相較於其他卷積神經網路更少，為輕量化模型。此模型大致可分為四階段，首先為一層全域卷積層，下一階段為兩次連續的深度可分離卷積，最後為一層全連階層兼輸出層，本專題具體之 ShuffleNet 模型架構如下表(一)所示：

表(一) 本專題 ShuffleNet 模型架構

| Layer | | Stride | Output Size | Output Channel |
|---|---|---|---|---|
| Input | | - | 64×64 | 1 |
| 3×3 Global Conv | | 2 | 32×32 | 16 |
| MaxPool | | 2 | 16×16 | 16 |
| Stage1 | Block1 | 2 | 8×8 | 32 |
| | Block2 | 1 | 8×8 | 32 |
| Stage2 | Block1 | 2 | 4×4 | 64 |
| | Block2 | 1 | 4×4 | 64 |
| | Block2 | 1 | 4×4 | 64 |
| | Block2 | 1 | 4×4 | 64 |
| Global AvgPool | | - | 1×1 | 64 |
| Fully Connected | | - | 2 | 1 |

B. 硬體端

硬體端通過外接麥克風實現音訊採集取得類比聲音訊號，再經由控制 FPGA 內配置的類比數位轉換器 Analog-to-Digital Converter (ADC)轉換為數位訊號；接著因頻譜訊號更利後續卷積神經網絡辨識，我們使用聲音預處理 MFSC 將時域訊號轉為頻域訊號；最後在狀態機與記憶體控制下，將即時的音頻特徵圖傳入 ShuffleNet 神經網路進行辨識，系統整體架構如圖(十一)。




圖(十一)車禍聲音識別整體硬體系統架構

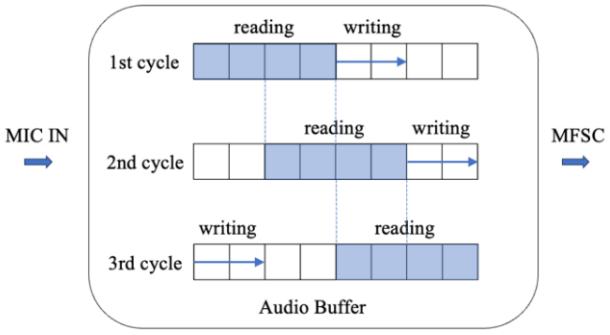
- 聲音訊號接收

本系統採用 FPGA 板通過內建音效卡外接麥克風與訊號放大器進行聲音訊號的接收，

並且藉由暫存區調整資料傳輸頻率來達成聲音訊號之即時傳輸。

本專題所使用之 FPGA(PYNQ-Z2) 自帶的音效卡為 ADAU 1761，內建 3.5mm 音效接口與 ADC，可以外接耳機、麥克風與處理類比訊號。在錄製音效的設備方面，我們考量了輕便性、指向性與頻率響應等因素，選擇了 audio-technica AT9901 麥克風作為收音設備。

聲音訊號接收後會被寫入於暫存區，供 MFSC 端進行讀取。為了提高傳輸效率，我們對聲音原始訊號暫存區進行了自行設計改良之乒乓操作，此操作之特色為可以對暫存區同時進行讀取與寫入。當系統開始運作後，聲音接收模組首先對暫存區寫入第一次傳輸週期所需的訊號量，準備完成後，MFSC 端便對暫存區進行第一個週期的資料讀取，同時進行第二個週期的資料寫入，以此類推，透過資料位移的方式，使暫存區在寫入第 n 週期的資料時，MSFC 可同時讀取第 n-1 週期被寫入的資料。此操作之另一重點為接續週期所傳輸的資料，會有部分重複，以保留聲音訊號之連續性。




圖(十二) 聲音訊號暫存區

- 聲音預處理

與軟體端介紹相同， MFSC 是根據人耳聽覺特性設計而成的非線性聲音預處理流程，其核心為梅爾過濾器。此部分與軟體端相似，依序由(i)加窗、(ii)快速傅立葉轉換與複數取平方、(iii)梅爾濾波以及(iv)取對數值構成。由於硬體端必須考慮記憶體的儲存容量限制，以及音訊的處理速度需達到「即時性」，也就是處理速度必須快於音訊資料的傳入速度，因此，我們必須簡化 MFSC 的流程以及梅爾濾波器組的參數精度，但仍需維持卷積神經網路的辨識準確度。

i. 加窗

此系統我們使用漢明窗，此為最常使用的窗函數之一，而硬體部分我們將軟體端產生的漢明窗數值先轉換為 16 位元定點小數的形式，再透過自訂 ROM IP 的方式存入 ROM 中，在聲音預處理階段，硬體會直接呼叫 ROM 提取漢明窗的參數並計算，提高運算效率。

ii. 快速傅立葉轉換與複數取平方

快速傅立葉轉換將時域音訊資料轉為頻譜以利卷積神經網路模型的音訊辨識。在硬體設計部分，快速傅立葉轉換在 vivado 中有現有的 IP 可以直接引用，因此，我們根據需求直接對 IP 做細部更改，改變其輸入輸出小數位數，以及選擇硬體量最小的設計。我們設定小數精度為 16 位元定點小數，與 ShuffleNet 卷積神經網路模型的參數精度一致，此部分將在卷積神經網路硬體實作部分詳細說明。接著，我們將複數的實部及虛部各自平方後相加，得到各頻率組成振幅的平方，加強頻譜的特徵，並削弱非特徵部分。硬體中我們直接使用 vivado 中的乘法器 IP，改變其輸入輸出的小數位數，確保計算結果正確。

iii. 梅爾濾波

梅爾濾波器是由多個不同中心頻率的脈波重疊而成，有助於去除特定頻率區域內的資料，從而獲得更具代表性的頻譜特徵。在設計梅爾過濾器時，為避免消耗過多硬體於梅爾濾波器組的初始化上，我們直接將軟體端產生的梅爾濾波器組先轉換為 16 位元定點小數的形式，再透過自訂 ROM IP 的方式存入 ROM 中，提高硬體的計算效率。接著，在聲音預處理階段，硬體會直接呼叫 ROM 提取濾波器組的參數並計算。

iv. 取對數值

為了要讓頻譜特徵更為明顯，我們對頻譜取其對數值，使特徵部分加強。此部分透過查表的方式達成。

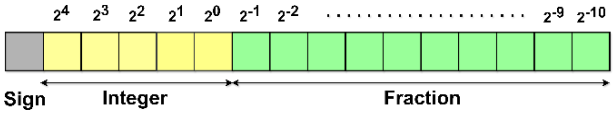
- ShuffleNet 神經網路與狀態機

我們的系統採用的是小型的 ShuffleNet 神經網路，包含了全域卷積層、兩個深度可分離卷積層以及一個全連接層。透過事先在

軟體端架設並訓練好神經網路，我們提取其參數載入到 FPGA 中。相較於傳統的卷積神經網路，ShuffleNet 在維持準度的情況下，擁有更少的計算量以及參數量，但相應代價更加複雜的計算步驟，因此需要設計對應的計算單元，並使用狀態機來調控每一步驟。

i. 16 位元定點數量化

傳統的神經網路運算通常是使用 32 位元的單精度浮點數運算，這代表著參數需要的儲存空間很大，且對於需要大量加法與乘法的神經網路來說，浮點數的緩慢計算速度也是個不容忽視的缺點。因此我們需要考慮在不減少過多精度的情況下，改變格式或是削減位數來提升運算效率以及減少需要的儲存空間。經過考慮我們決定採用 16 位元的定點數運算，分成了 1 位符號位 (Sign)、5 位整數 (Integer)及 10 位小數(Fraction)，如圖(十三)。




圖(十三) 16 位元定點數分配

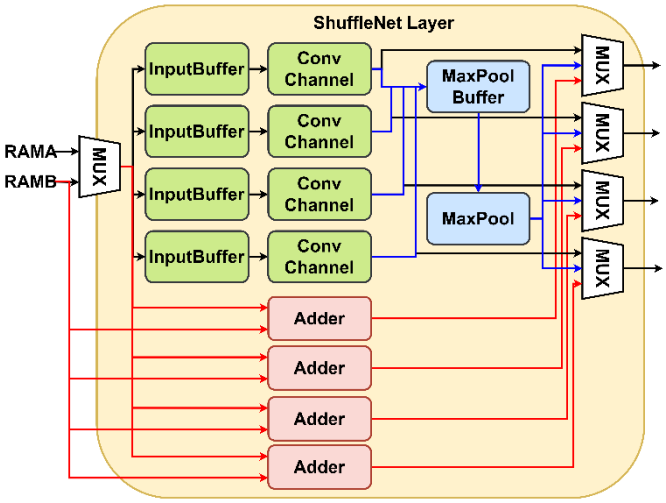
ii. ShuffleNet 神經網路

ShuffleNet 需要的計算步驟有很多種類，大致可以分為以下幾種，如表(二):

表(二) ShuffleNet 計算步驟與種類表

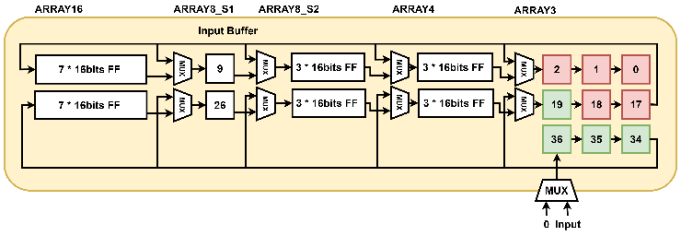
| 步驟類型 | 說明 | 種類 |
|---|---|---|
| DepthWise Conv1 | 步長 1 深度卷積 | DW |
| DepthWise Conv2 | 步長 2 深度卷積 | DW |
| PointWise Conv | 逐點卷積 | PW |
| MaxPool3*3 | 3*3 最大池化 | - |
| AvgPool2*2 | 2*2 平均池化 | PW |
| AvgPool4*4 | 4*4 平均池化 | PW |
| Add | 兩通道相加 | ADD |
| Fully Connected | 全連接層 | PW |

由於計算步驟種類繁多，為每一種步驟類型都各自設計一套電路顯然不利於輕量化。為此我們重新分析了各種計算步驟的性質，將逐點卷積、平均池化以及全連接層這些前後兩次計算不會出現輸入資料重疊的類型分為 PW；會出現資料重疊的深度卷積分為 DW；兩通道相加的部分因為每次只需一筆資料，因此單獨拉出來分為 ADD；而最大池化較為不同，由於初始的全域卷積輸出資料非常龐大，且與最大池化一起佔用了將近四分之一的時間，因此我們對其優化，做成流水線來減少時間與空間使用。以下是整體計算層的結構:




圖(十四) 計算層(ShuffleNet Layer)結構圖

計算層由四個平行的計算通道組成，這有助於提高運算速度。考慮到電路複用的情況，PW、DW 本質上都是相乘相加的卷積操作，其資料路徑為上圖中的綠色區塊，從輸入端進來依序是輸入緩衝、卷積通道；接在藍色部分則是最大池化的緩衝與計算單元；ADD 則是走紅色路徑使用加法器進行相加。

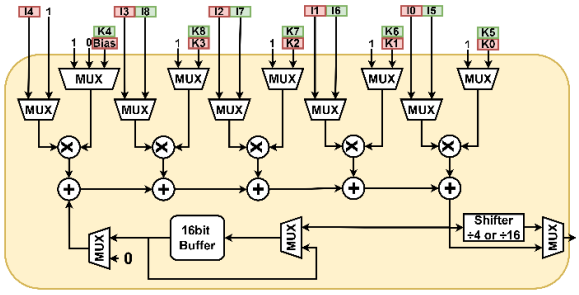



圖(十五) 輸入緩衝(InputBuffer)結構圖

圖(十六) 輸入緩衝(InputBuffer)數據流圖

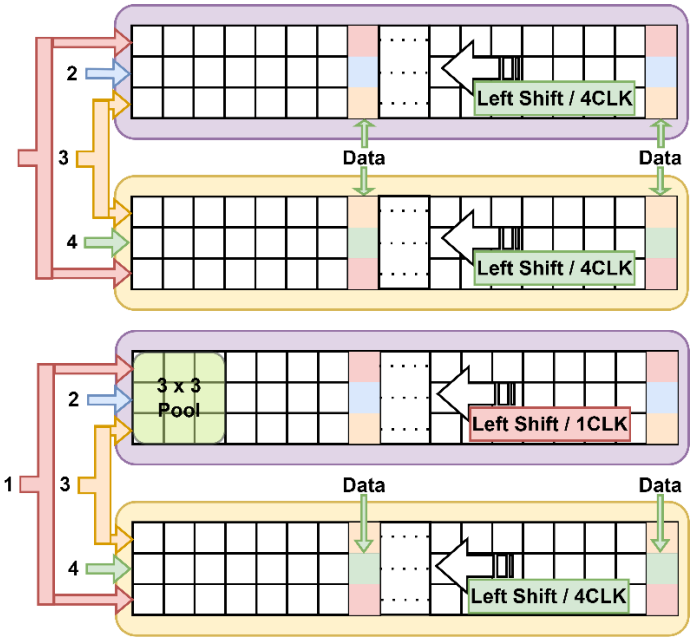
考量到 RAM 輸出位寬不足及需要處理 DW 這類輸入資料重疊的情況，我們使用移位暫存器做為緩衝。由於存在多種不同大小的特徵圖，有些還包含 Padding，因此我們利用 tapped delay line 的方式，容許緩衝以多條線路移位。

圖(十六)為 4*4 卷積(含 Padding)時，前四個計算周期使用 RAM 位置，黃色為緩衝當前輸出，綠色為緩衝當前輸入，橘色為暫存數據。




圖(十七) 卷積通道(ConvChannel)結構圖

卷積通道由五組乘法-加法器、移位器、多工器與激活函數 ReLU6 組成，並包含 16bits 暫存器存放暫時資料。此設計容許電路在 2 個 Clk 下計算 3*3 的卷積以及額外一位 Bias。透過輸入端的多工器轉換輸入類型，我們得以用卷積通道計算平均池化。同時，輸出端的移位器是用於處理平均池化的除法。為了減少資源使用量，特別使用 2*2 與 4*4 兩種平均池化而不是更常見的3*3，這允許我們利用簡單的右移2位或 4 位來實現原本需要除法器的除以 4 或 16。透過狀態機調控多工器的輸出，我們得以將不同類型的操作都在卷積通道實現。

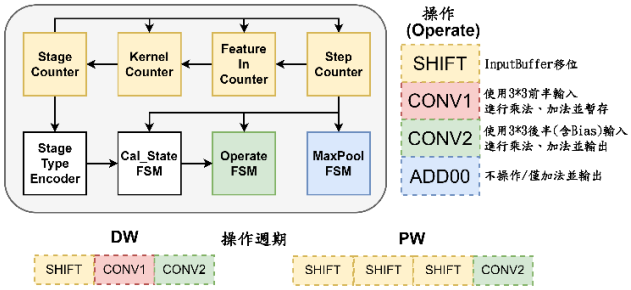



圖(十八) 最大池化緩衝(MaxPool Buffer)結構圖

若要將全域卷積與最大池化做流水線，要考慮兩者存取不能衝突，因此設計成 Ping-Pong 形式並拆成四個狀態，狀態 1、3 會同時寫入兩組緩衝，上面的圖表達的是狀態 3(橘色箭頭)；狀態 2、4 則是寫入一組、池化另一組，下面的圖表達的是狀態 4(綠色箭頭)，此時上半緩衝進行池化，而下半維持寫入緩衝。

iii. ShuffleNet 狀態機

狀態機由 Stage、卷積核、特徵圖座標、Step 等四個計數器主導(黃色部分)，前一計數器達目標值後就會清零並驅動下一級計數器，如此遍歷所有步驟。綠色部分是操作狀態機，由各計數器數值可推得當下操作(Operate)種類，透過分析所有的計算都能拆為四種操作(圖右半)，能簡化控制訊號實作，例如圖下半 DW 與 PW 兩種操作週期。藍色部分則是控制最大池化流水線的狀態機。

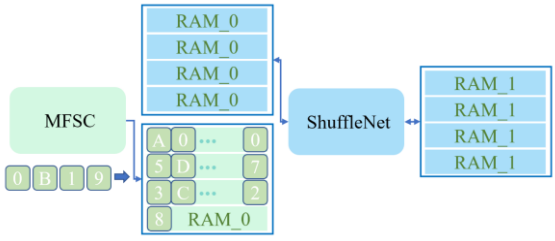



圖(十九) ShuffleNet 狀態機設計圖

- 記憶體控制

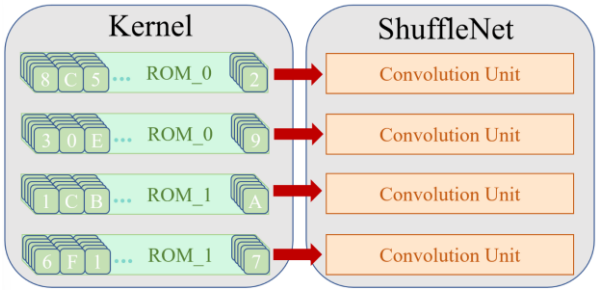
為達成邊緣運算所需的輕便性，記憶體端的輕量化可分為兩部分，從模型設計角度上，ShuffleNet 使用 pointwise、depthwise 和 shuffle 等運算已大幅減少參數需求量；接著硬體端的設計存取上，我們也盡可能使用最少記憶體的前提下達成空間重複利用。

記憶體控制端介於整體系統的聲音預處理 MFSC 模組與 ShuffleNet 運算模組之間。因此，硬體設計可分為以下兩個課題：其一、即時聲音頻譜數據的暫存；此部分涉及 Ping-Pong RAM 設計，需動態切換與聲音預處理模組 MFSC 及 ShuffleNet 運算所連接之記憶體區塊，如圖(二十)。




圖(二十) 記憶體控制 RAM 概念圖

其二、ShuffleNet 運算所需之 kernel 參數存取；此部分會將 ShuffleNet 訓練出的參數以 coe 檔格式匯入 FPGA 唯讀記憶體 ROM 中，並依序取出參數與頻譜數據運算，如圖(二十一)。




圖(二十一) 記憶體控制 ROM 概念圖

本系統共有 4 通道 ShuffleNet 運算單元，因此 FPGA 所搭載之記憶體單元 BRAM 也是以四個獨立 BRAM 合成為一組，所以此部分共使用 12 個獨立的 BRAM IP，位寬皆為 16bits。而在 kernel 儲存端亦是 4 個 ROM IP 與 4 通道進行對應，且因為 convolution unit 設計是一個 clock 執行 5 個運算，故 kernel 輸出位寬是 5 個參數(80bits)，下表(三)為記憶體空間使用狀況。

表(三) 記憶體控制端用量紀錄表

| 記憶體種類 | 位寬(bit) | 深度(個) | 數量(個) |
|---|---|---|---|
| RAM_0 | 16 | 1088 | 8 |
| RAM_1 | 16 | 1024 | 4 |
| ROM_0 | 80 | 324 | 2 |
| ROM_1 | 80 | 308 | 2 |

在表(三)中 RAM_0 是聲音預處理 MFSC 模組與 ShuffleNet 運算模組之間使用的 Ping-Pong RAM，當其中四個存取來自 MFSC 的資料時，另外四個就會與 RAM_1 共同作為 ShuffleNet 運算時的暫存區。ROM_0 和 ROM_1 都是用於儲存所需的 Kernel 參數，深度不同是由於全連接層只會有"車禍"和"非車禍"兩種結果，因此只需要用 ROM_0 儲存深度 16 的全連接層參數。

# V. 實驗結果

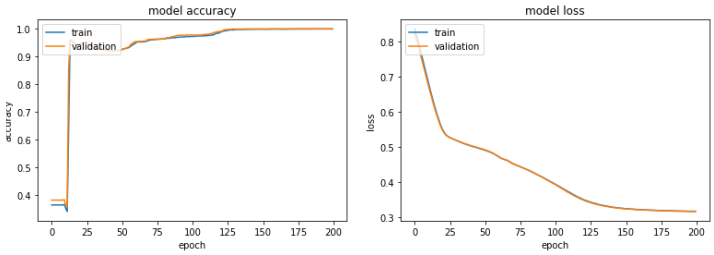
## A. 軟體端

訓練集與測試集資料總數如下表所示，我們將原先蒐集的音訊資料集進行 Data augmentation，提高模型辨識準確率。

表(四) 資料集分布表

| 音訊 | 車禍事件 | 非車禍事件 | 總數 |
|---|---|---|---|
| 訓練集 | 4158 | 2419 | 6577 |
| 驗證集 | 1386 | 806 | 2192 |
| 測試集 | 1851 | 1076 | 2927 |

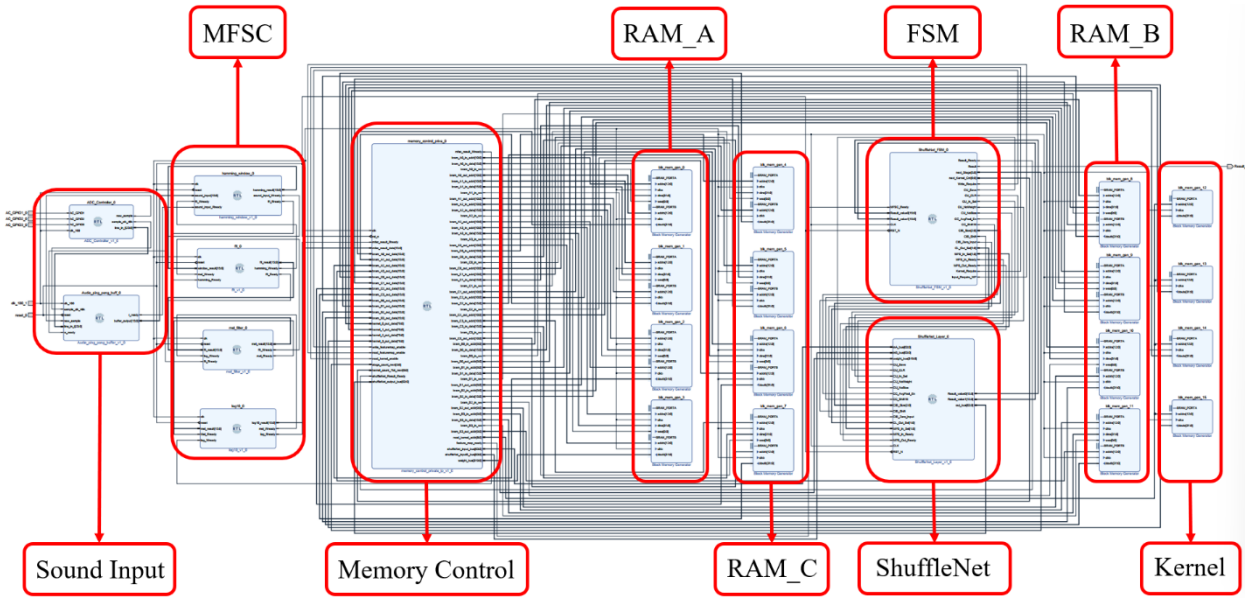
ShuffleNet 模型訓練及驗證過程的準確率與損失函數如下圖(二十二)所示，我們使用 categorical crossentropy 作為 loss 函數，並用 Adam 做選擇器，測試集得到的準確率為 92.7%，F1 score 為 93.9%，loss 為 0.378。




圖(二十二) ShuffleNet 訓練結果圖

## B. 硬體端

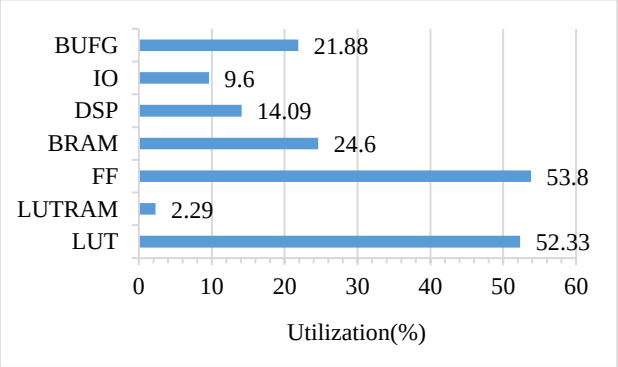
硬體端系統整體系統架構如圖(二十三)。實作上，本專題採用 PYNQ-Z2 型號 FPGA，以下表(五)為系統各部分硬體使用量，而圖(二十四)為總系統資源使用占比。




圖(二十三) 硬體系統整體架構圖

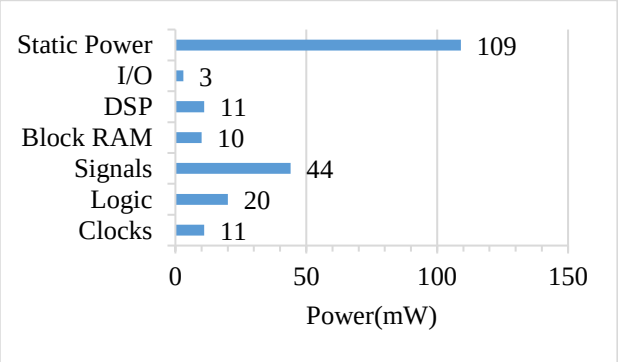
表(五) 系統各部分硬體使用表

| | Sound Input ~MFSC | Memory Control ~ShuffleNet |
|---|---|---|
| LUT | 20672 | 6061 |
| LUTRAM | 110 | 192 |
| Flip-Flop | 50982 | 4451 |
| BRAM | 17 | 17 |
| DSP | 7 | 24 |




圖(二十四) 整體系統資源比例圖

本專題之系統在滿足硬體資源輕量化的同時，也盡量追求低功耗的邊緣運算特性，如圖(二十五)呈現整體系統的功率消耗。




圖(二十五) 整體系統功耗分布圖

## VI. 討論與結論

在本專題中，我們致力於開發一套道路聲音辨識的邊緣智慧系統，此系統結合軟硬體以達成高輕量化與高準確度的車禍聲音識別。開發過程可概述為以下 3 個主要步驟：

1. 資料集建立: 收集並整理用於訓練和驗證模型的聲音數據，其中包括真實車禍聲音以及人造音效等；並進行資料集平衡與增強，以建立一個全面的車禍聲音資料庫。
2. ShuffleNet 神經網絡模型訓練:對預處理過的聲音數據應用 ShuffleNet 進行學習和模型訓練；在維持高準確度的辨識同時，盡可能降低硬體模型的參數量與運算複雜度。
3. 硬體端開發: ShuffleNet 網絡架構透過平行運算、流水線以及硬體複用等技術，搭配狀態機與記憶體控制模組，成功實現輕量化的車禍聲音辨識系統。
4. 軟硬體整合: 將軟體端的聲音預處理 MFSC 模組與 ShuffleNet 神經網絡實作在 FPGA 上，不僅提高整體系統的運算速度和效能，也成功達成實時的車禍聲音辨識邊緣系統。

本專題所提出之技術不僅有潛力彌補傳統攝影機的視覺盲點，也可顯著加快對緊急情況的反應速度，展現了監視器系統功能的強化潛力。同時解決硬體資源的限制問題，提供了一個既實用又可行的方案。未來，我們希望將影像辨識功能整合進本系統中，透過聲音與影像的結合，我們能更全面、更準確地偵測交通事故，為緊急救援工作贏得關鍵的時間。

## VII. 參考文獻


[1] 蕭依娜, "針對非特定語者語音辨識使用不同前處理技術之比較," 碩士論文, 電控工程研究所, 國立交通大學, 新竹, 2003, pp. 75.
[2] Kaur, S., "Mouse Movement using Speech and Non-Speech Characteristics of Human Voice," International Journal of Engineering and Advanced Technology, Jan. 2012.
[3] Z. K. Abdul and A. K. Al-Talabani, "Mel Frequency Cepstral Coefficient and its Applications: A Review," in IEEE Access, vol. 10, pp. 122136-122158, 2022
[4] Dua, M., et al., "Speaker Recognition Using Noise Robust Features and LSTM-RNN," in Progress in Advanced Computing and Intelligent Engineering, pp. 19-28, Apr. 2021
[5] L. Alzubaidi, J. Zhang, A. J. Humaidi et al., "Review of deep learning: concepts, CNN architectures, challenges, applications, future directions," in Journal of Big Data, vol. 8, no. 1, pp. 53, 2021.
[6] K. M. Tarwani and S. Edem, "Survey on recurrent neural network in natural language processing," in International Journal of Engineering Trends and Technology, vol. 48, no. 6, pp. 301-304, June. 2017.
[7] Y. Kim, "Convolutional Neural Networks for Sentence Classification," in Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP 2014), pp. 1746-1751, Aug. 2014.
[8] W. Fang, P.E.D. Love, H. Luo, L. Ding, "Computer vision for behaviour-based safety in construction: a review and future directions," in Advances in Engineering Informatics, vol. 43, pp. 100980, Aug. 2019.
[9] D. Palaz, M. Magimai-Doss, R. Collobert, "End-to-end acoustic modeling using convolutional neural networks for HMM-based automatic speech recognition," in Speech Communication, vol. 108, pp. 15-32, Apr. 2019.
[10] Y. Lecun, L. Bottou, Y. Bengio and P. Haffner, "Gradient-based learning applied to document recognition," in Proceedings of the IEEE, vol. 86, no. 11, pp. 2278-2324, Nov. 1998.
[11] Howard, A.G., et al., "Mobilenets: Efficient convolutional neural networks for mobile vision applications," in arXiv preprint arXiv:1704.04861, Apr. 2017.
[12] X. Zhang, X. Zhou, M. Lin and J. Sun, "ShuffleNet: An Extremely Efficient Convolutional Neural Network for Mobile Devices," 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 2018, pp. 6848-6856
[13] P. Peng et al., "Design of an Efficient CNN-Based Cough Detection System on Lightweight FPGA," in IEEE Transactions on Biomedical Circuits and Systems, vol. 17, no. 1, pp. 116-128, Feb. 2023
[14] S. R. Shahamiri, "Speech Vision: An End-to-End Deep Learning-Based Dysarthric Automatic Speech Recognition System," in IEEE Transactions on Neural Systems and Rehabilitation Engineering, vol. 29, pp. 852-861, 2021
[15] G. Valenzise, L. Gerosa, M. Tagliasacchi, F. Antonacci and A. Sarti, "Scream and gunshot detection and localization for audio-surveillance systems," 2007 IEEE Conference on Advanced Video and Signal Based Surveillance, London, UK, 2007, pp. 21-26
[16] M. Cristani, M. Bicego and V. Murino, "Audio-Visual Event Recognition in Surveillance Video Sequences," in IEEE Transactions on Multimedia, vol. 9, no. 2, pp. 257-267, Feb. 2007
[17] S. Rovetta, Z. Mnasri and F. Masulli, "Detection of Hazardous Road Events From Audio Streams: An Ensemble Outlier Detection Approach," 2020 IEEE Conference on Evolving and Adaptive Intelligent Systems (EAIS), Bari, Italy, 2020, pp. 1-6
[18] F. Iqbal, A. Abbasi, A. R. Javed, G. Srivastava, Z. Jalil and T. R. Gadekallu, "Identification and Categorization of Unusual Internet of Vehicles Events in Noisy Audio," 2023 IEEE 97th Vehicular Technology Conference (VTC2023-Spring), Florence, Italy, 2023, pp. 1-6
[19] Z. Wang, W. Zha, J. Chai, Y. Liu and Z. Xiao, "Lightweight Implementation of FPGA-Based Environmental Sound Recognition System," 2021 International Conference on UK-China Emerging Technologies (UCET), Chengdu, China, 2021, pp. 59-66
[20] P. Foggia, N. Petkov, A. Saggese, N. Strisciuglio and M. Vento, "Audio Surveillance of Roads: A System for Detecting Anomalous Sounds," in IEEE Transactions on Intelligent Transportation Systems, vol. 17, no. 1, pp. 279-288, Jan. 2016
[21] Sammarco, M. and M. Detyniecki. "Crashzam: Sound-based Car Crash Detection," in 4th International Conference on Vehicle Technology and Intelligent Transport Systems, Mar. 2018.
[22] T. -V. Şerban-Moga, L. Grama and C. Rusu, "Classification and Identification of Certain Types of Car Accidents Based on Sound Information," 2023 International Conference on Speech Technology and Human-Computer Dialogue (SpeD), Bucharest, Romania, 2023, pp. 30-35
[23] Piczak, K. J. (2015). "ESC: Dataset for Environmental Sound Classification," Proceedings of the 23rd ACM international conference on Multimedia, Oct. 2015, pp.1015-1018
[24] Youtube Channel: Car Crashes Time: https:/www.youtube.com/channel/UCil5Tyte_KTTrPgt5cC5Qw/videos
[25] Sundsnap--Car Crash Sound Effects: https//www.soundsnap.com/tags/car_crash
[26] Sorten, C., Khoshgoftaar, T.M. "A survey on Image Data ugmentation for Deep Learning, " in Journal of Big Data 6, 60, July. 2019
[27] Islam, Z., & Abdel-Aty, M. A. "Deep Convolutional Neural Network for Roadway Incident Surveillance Using Audio Data, " in ArXiv, abs/2203.06059, Mar. 2022