

國立交通大學

電子工程學系 電子研究所碩士班

碩 士 論 文

助聽器的噪音消除演算法



Noise cancellation algorithm in hearing aids

研 究 生：鍾 譯 賢

指導老師：桑 梓 賢 教授

中 華 民 國 九 十 九 年 三 月

助聽器的噪音消除演算法

Noise cancellation algorithm in hearing aids

研 究 生：鍾譯賢

Student: Yi-Hsien Chung

指導教授：桑梓賢 教授

Advisor: Tzu-Hsien Sang

國 立 交 通 大 學

電子工程學系 電子研究所碩士班



Submitted to Department of Electronics Engineering & Institute of Electronics

College of Electrical and Computer Engineering

National Chiao Tung University

in Partial Fulfillment of the Requirements

for the Degree of

Master

in

Electronics Engineering

March 2010

Hsinchu, Taiwan, Republic of China

中 華 民 國 九 十 九 年 三 月

助聽器的噪音消除演算法

研究生：鍾譯賢

指導教授：桑梓賢 教授

國立交通大學
電子工程研究所碩士班

摘要

子空間演算(subspace method)，頻譜相減演算(subtraction method)與卡爾曼濾波器(Kalman filter)等在處理語音方面已使用多年，在消雜訊方面也有其效果，所以期望能將其效果使用在助聽器上，來得到較好語音的訊號，使助聽器在使用上也能擁有較好的語音品質。

為因應助聽器的需要，演算法須滿足一些特性，一 計算量，處理過程不能太大或是複雜，因為助聽器在使用上，無法容忍處理時間過長的情況，二 低功率，因為使用者幾乎是全天配帶，若是需要常常更換電池或充電，將會大大降低其實用性。我們在計算量方面，盡可能尋找結構較簡單的演算法，或是將演算法的計算加以簡化，低功率方面，則是盡量善用濾波器組(filter bank)帶來的一些好處，例如硬體共用，分頻取代 DFT 計算等等，另外在使用濾波器組的架構下，有時也會帶來提升演算法效果的機會，期望在運算複雜度與效果之間能取得一個最佳的平衡。

以下為章節排序，章節一為消雜訊演算法的相關工作，主要介紹一些語音的客觀評估方法，章節二到五，為演算法的介紹，相關的演算法有子空間演算，頻譜相減演算，卡爾曼(Kalman filter)以及雙耳演算，章節六為濾波器組的硬體架構介紹，章節七跟八則是將適用的演算法應用到濾波器組上，章節九則為一些演算法的特性比較與結論。

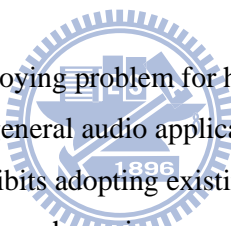
Noise cancellation algorithm in hearing aids

Student : Hsien Chung

Advisor : Tzu-Hsien Sang

**Department of Electronics Engineering & Institute of Electronics
National Chiao Tung University**

ABSTRACT



Noise amplification has been an annoying problem for hearing aid user. There are several effective noise reduction algorithms for general audio applications. But for hearing aids, the requirement of real-time processing prohibits adopting existing approaches with high computation complexity. In this paper, a noise reduction scheme is proposed to utilize the filter bank structure which is already required for the function of hearing-loss compensation. Through such hardware-sharing arrangement, it is hopeful to achieve low hardware and, most importantly, real-time noise reduction.

誌謝

其實我想整篇論文裡，最值得看的段落就是這裡了，因為相較於研究結果，研究過程會比較容易讓我想起曾經發生過甚麼，而講到過程就沒法不去題在交大最後兩年受到的幫忙與照顧。

首先要謝謝桑老師對我的包容與耐心，容忍我笨拙的程式撰寫能力，謝謝教授在討論的過程中，讓我體悟到表達與描述的重要，要怎麼呈現才會有表達的效果，以及分析事情的觀點，在遇到陌生議題時，該把注意力放那，沒讓我耗費力氣在無謂的事上，使我可以順利完成研究論文。

再來要謝謝哲聖跟宗達，謝謝你們陪我度過無數個在實驗室的夜晚，這句話其實我想了半年，打開電腦寫的時候又想了半小時，最後好像還是沒法再多寫些甚麼，再寫筆就停不下來了。

黃俊豪跟曾鉉鈞，因為他們笨笨的相信與支持，所以我又回到交大唸書。

正湟學長，謝謝由始至終一直提供很大的協助，尤其在是論文或書籍方面。

誠文學長，在計畫初期就參與，謝謝你陪我一起走到很多地方，能與你分享許多研究中的發現。

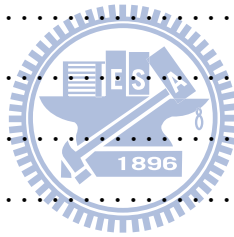
欣德學長，謝謝學長以自身經驗傳承，讓我在做一些事情的時候有所依據，以及在論文寫作方面提醒我很多有關架構與細節的問題。

電物徐琅教授與弘神父，他們影響我甚多，讓我對於生活諸多無奈與不可避免能有所接受與釋懷，雖說非研究範疇，但有助我將心思聚焦在研究上。

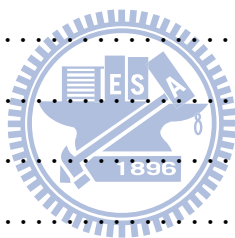
最後還有已畢業的聰文跟智維學長，在寫程式初期給予協助，以及供我使役的學弟旭謙與耀賢，謝謝你們的任勞任怨，還有最後半年階段的郁婷, 大 Q, 偵源及兆軒，讓我一直有機會對一些基本學科不致太過生疏以及在 LAB 長時間的陪伴。

目錄

中文提要	i
英文提要	ii
誌謝	iii
目錄	iv
表目錄	vi
圖目錄	vii
第一章 緒論與相關工作	1
1.3 語音偵測	3
1.5 章節概要	8
第二章 演算法	9
2.1 子空間演算	9
2.1.1 子空間原理介紹	9
2.1.2 子空間演算語音評比	11
2.2 頻譜相減演算	12
2.2.1 頻譜相減演算原理介紹	12
2.2.2 子空間演算語音評比	14
2.3 卡爾曼演算	15
2.3.1 卡爾曼演算原理介紹	15
2.3.2 卡爾曼演算語音評比	18
2.4 雙耳演算	20
2.4.1 雙耳演算原理介紹	20
第三章 濾波器組與頻譜相減演算應用	25
3.1 濾波器組	25
3.1.1 濾波器組設計規格	25



3.1.2 濾波器組的副作用	27
3.1.3 54 頻帶濾波器組設計	28
3.2 頻譜相減演算應用	31
3.2.1 頻譜相減演算的分頻設計	31
3.2.2 頻譜相減演算應用至濾波器組	31
3.2.3 權重更新方法	34
3.3 頻譜相減演算應用之語音評比	38
3.3.1 頻譜相減演算應用至 54 頻帶語音評比	38
3.3.2 頻譜相減演算應用至 18 頻帶語音評比	40
第四章 濾波器組與卡爾曼演算應用	42
4.1 卡爾曼演算的分頻設計	42
4.2 卡爾曼演算應用至 18 頻帶語音評比	44
第五章 結論與展望	47
5.1 結論	47
5.2 演算法回顧與比較	48
5.3 展望	50
參考文獻	51



表目錄

表 2.1	子空間演算 PESQ	14
表 2.2	子空間演算 SNRseg	11
表 2.3	頻譜相減演算 PESQ	14
表 2.4	頻譜相減演算 SNRseg	15
表 2.5	卡爾曼演算 PESQ	18
表 2.6	卡爾曼演算 SNRseg	18
表 2.7	雙耳演算的優勢與代價	24
表 3.1	濾波器組的頻率規範	27
表 3.2	頻譜相減演算 54 頻帶 PESQ	38
表 3.3	頻譜相減演算 54 頻帶 SNRseg	39
表 3.4	頻譜相減演算 18 頻帶 PESQ	40
表 3.5	頻譜相減演算 18 頻帶 SNRseg	40
表 4.1	卡爾曼演算 18 頻帶 PESQ	44
表 4.2	卡爾曼演算 18 頻帶 SNRseg	45
表 5.1	演算法優缺點比較	48
表 5.2	演算法摘要	48
表 5.3	演算法應用的優缺點比較	49
表 5.4	演算法應用的特性比較	49
表 5.5	頻譜相減演算 18 頻帶的計算量	50
表 5.6	卡爾曼演算 18 頻帶的計算量	50

圖目錄

圖 1.1	VAD 的判別結果	5
圖 1.2	SNRseg 的計算過程	7
圖 2.1	子空間演算流程圖	10
圖 2.2	子空間演算 PESQ	11
圖 2.3	子空間演算 SNRseg	12
圖 2.4	頻譜相減演算流程圖	14
圖 2.5	頻譜相減演算 PESQ	14
圖 2.6	頻譜相減演算 SNRseg	15
圖 2.7	卡爾曼演算流程圖	17
圖 2.8	卡爾曼演算 PESQ	18
圖 2.9	卡爾曼演算 SNRseg	19
圖 2.10	雙耳演算模型	21
圖 2.11	雙耳演算流程圖	23
圖 3.1	濾波器組結構圖	25
圖 3.3	濾波器的頻率響應	27
圖 3.4	18 頻帶濾波器組	28
圖 3.5	54 頻帶濾波器組	29
圖 3.6	18 頻帶與 54 頻帶濾波器組的頻率響應	30
圖 3.7	頻譜相減演算應用至濾波器	32
圖 3.8	頻譜相減演算與濾波器組結構圖	33
圖 3.9	權重更新流程圖	35
圖 3.10	權重更新 1	37
圖 3.11	權重更新 2	38
圖 3.12	頻譜相減演算 54 頻帶 PESQ	39

圖 3.13 頻譜相減演算 54 頻帶 SNRseg	39
圖 3.13 頻譜相減演算 18 頻帶 PESQ	40
圖 3.14 頻譜相減演算 18 頻帶 SNRseg	41
圖 4.2 卡爾曼演算結構圖	43
圖 4.3 卡爾曼演算 18 頻帶 PESQ	45
圖 4.4 卡爾曼演算 18 頻帶 SNRseg	45



第一章 緒論與相關工作

1.1 研究方向

在助聽器研究計畫中，會依照患者受損的聽力區塊給予補償，但在給予補償的同時也會放大語音訊號之外的雜訊，所以只是將聲音放大，並沒有在辨識度上有所增加，所以為了達到在語音補償後能有較好的語音辨識度，可以將語音訊號之外的訊號給予某程度的消除或是壓抑，藉此來突顯語音訊號。

助聽器遇到的雜訊可分為背景雜訊與回音雜訊，背景雜訊的定義是，在助聽器接受訊號前，除了語音訊號之外的訊息就算是背景雜訊，而回音雜訊則是由助聽器本身的結構所造成，因為要使內外耳壓力一致，所以會保留一通道，而內耳的擴音器也會經由此通道將過去的訊號傳回接收端，所以回音雜訊的定義是，處理過訊號經通道回饋致接收端的訊息稱之，以下則有附圖說明。

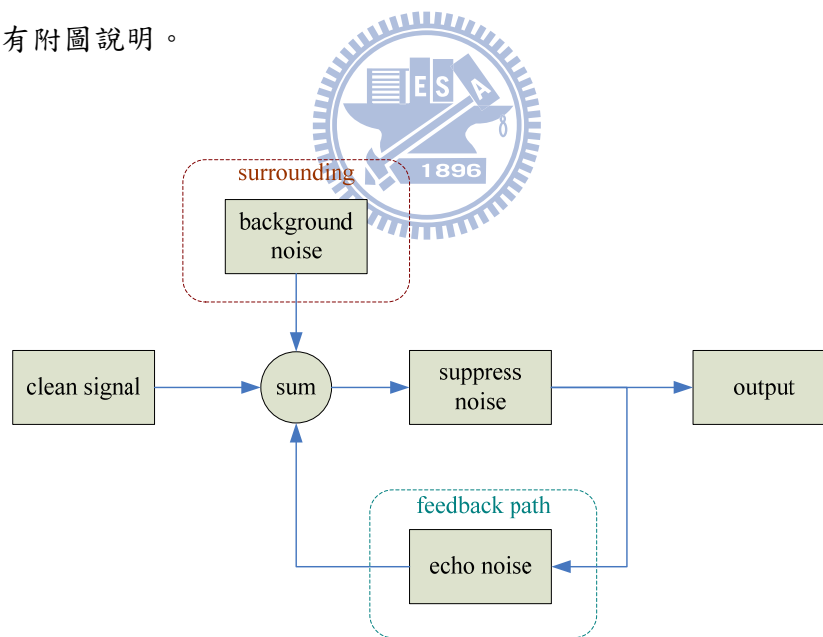


圖 1.1 背景雜訊與回音雜訊示意圖

至於提升語音辨識度的方法，則不外乎使用演算法來壓抑或消除雜訊，所以研究了幾種可以強健語音的演算法，目前涉及到的演算法有子空間演算,頻譜相減演算,卡爾曼和雙耳演算等。在初步的研究中只將注意力放在演算法對語音辨識上有多少貢獻，往後則會再對複雜度,計算量,功率及晶片面積大小等能否實作的議題進行討論。(方法)

在語音補償方面，則是將人類聽覺頻率範圍切分成 18 個區塊，又因語音較重要的訊息集中在低頻，所以低頻切分區塊較細，這樣的硬體架構濾波器組已經由其他實驗室完成，而演算法在使用的時候則需做某程度的修改，來遷就這樣的硬體架構，降低對複雜度、計算量、功率及晶片面積大小的需求，提高實作的可能性。

1.2 演算法簡介

子空間演算的主要想法是將污染語音投影至雜訊子空間與語音子空間，如果可以確實將污染語音區分成雜訊子空間與語音子空間，那就盡可能壓低污染語音在雜訊子空間的分量，保留污染語音在語音子空間的分量，來達到消雜訊的效果。

頻譜相減法演算的構想是，將污染語音轉至到頻域上，污染語音在頻率上的分量大小為語音與雜訊的總合，若可以知道語音與雜訊在頻率上的分量大小，則可以將污染語音在頻率上的分量還原至原來語音在頻率上的分量大小，進而達到消雜訊的目的。

卡爾曼演算，係利用現下語音與過去語音具有穩態特性，選用有效的過去資料為基底，將這些基底經線性組合來預測下一刻的語音，進而對污染語音進行還原。卡爾曼演算與自迴歸過程(Autoregressive process)極為類似，不同的地方在於基底選取，自迴歸過程在選取基底上是直接使用過去資料為基底，而卡爾曼演算則較為複雜一些，因為輸入訊號為污染訊號，若直接使用過去資料為基底較難拼湊出原語音，故尚需估計出背景雜訊，然後再對過去資料加以修正才能當作基底使用。

雙耳演算法的想法是，利用語音訊號到達左右耳時間相同，而雜訊到達左右耳的時間不同，具有一個時間差，因左右耳皆有個別的接收器，將左右耳接收的訊號相減作傅立葉轉換可以得到較完整的背景雜訊，在提供較完整的背景雜訊下，上述的各演算法可以更趨於理想，提升對污染語音消雜訊的能力。

1.3 語音偵測

語音偵測(voice active detection VAD)的功用，用在判斷汙染語音中，那段為語音區間，那段為非語音區間，這是大多演算法都免不了要使用的工具，非語音區間內含有純雜訊的資訊，而語音區間則是同時夾雜了語音跟雜訊，一般而言演算法都依照從得到的資訊，才能來對的訊號進行處理。

1.3.1 振幅強度法

振幅強度法(Amplitude method)的想法相當直觀而且簡單，一般看到一段語音，其語音區間與非語音區間最大的差異應該就是振幅，也就是語音區間擁有較大的能量，所以對任意一段汙染語音計算其振幅平方總合，就很容易判斷出其差異。

汙染語音一開始，通常為非語音區間，所以往後未知的區間只需與一開始的非語音區間比較其振幅平方總合，就可以判別未知區間為語音區間或非語音區間，而最初語音區間和非語音區間的振幅平方總合比，則可視為一個門檻值，當作往後的一個比較基準。

振幅強度法的優點是想法簡單容易實踐，缺點則是對門檻值頗為敏感，所以門檻值設定的恰當與否，對於判定結果具有相當大影響。

1.3.2 Teager 能量算子法

Teager 能量算子法(Teager method)的想法是利用訊號在時間上的相關性，一般而言語音區間和非語音區間在時間上的相關性高低有所不同，來辨別是否為語音訊號至於量化未知區間為相關程度，則藉由 Teager 能量算子(teager energy operator,TEO)來計算，其定義如下

$$\psi[s(n)] = s(n)^2 - s(n-1)s(n+1) \quad (1.1)$$

s 為汙染語音

(2.1)式右邊的第一項，是在計算現下訊號的相關程度，因為與現下訊號最相似的就是其訊號本身，右邊的第二項，則是計算前一刻與下一刻訊號的相關程度，如果訊號本身相關性就高，則第一項與第二項的值都會較大且較相近，所以 TEO 值會較小，若訊號相關性低，則會有第一項較大而第二項較小的情況，相對 TEO 值就會大。

Teager 能量算子是針對一個離散的點來作用，但語音是一堆點群造成的效果，所以需要對語音觀察一段時間才知道此語音區間相關性高低為何，故對 TEO 取了期望值，其式如下

$$E\{\psi[s(n)]\} = E\{s(n)^2\} - E\{s(n-1)s(n+1)\} \quad (1.2)$$

Teager 能量算子法的優點與振幅強度法類似，就是想法跟結構都不算複雜，容易實作，缺點是，計算量，在乘法次數與加法次數方面，較振幅強度法各多出一倍，另外 Teager 能量算子法也有類似振幅強度法的缺點，就是門檻值的設定，Teager 能量算子法也需要定出一個門檻值，超過判為非語音區間，未過判為語音區間，所以門檻值定得是否恰當，會是判斷能否準確的主因。

1.3.3 亂度法

亂度法(Entropy method)的想法是利用訊號在頻譜上的分佈特性，來加以區分是否為語音區間，一般而言語音區間和非語音區間在頻譜上的分佈情況有所不同，人類語音大部分集中在低頻，所以低頻功率占總功率絕的大部分，雜訊在頻譜上的分佈，則應該是比較隨機。

辨別的方式就是，如果有未知區間其功率分佈情況類似人類語音，即可判為是語音區

間，接著就是量化未知區間與人類語音功率分佈的相似度，首先要先計算未知區間在每個頻帶功率的分佈大小及其百分比，計算方式如下

$$p_i = s(f_i) / \sum_{k=1}^N s(f_k) \quad (1.3)$$

$s(f)$ 為訊號在某頻帶上的功率， N 表示將整個頻域切成多少個頻帶， i 與 k 皆為頻帶的編號。由(2.3)式，可以得知，未知語音區間功率分佈在各個頻帶上的百分比，即可透過亂度計算以及權重(weight)的調整，來量化頻譜相似的程度

$$H = -\sum_{k=1}^N w_k p_k \log p_k \quad (1.4)$$

W 為權重， H 為亂度計算

亂度法的好處是，它只注重頻譜上功率分佈情況，所以在低 SNR 的情況下精準度比較不受影響，但缺點則如同其他 VAD，亂度法也是需要一個門檻值來當作相似程度的準則，另外在計算量上跟複雜度上也是超過 Teager 能量算子法和振幅強度法許多，以下為 VAD 判定語音的結果，紅線 HIGH 的部分為語音區間。

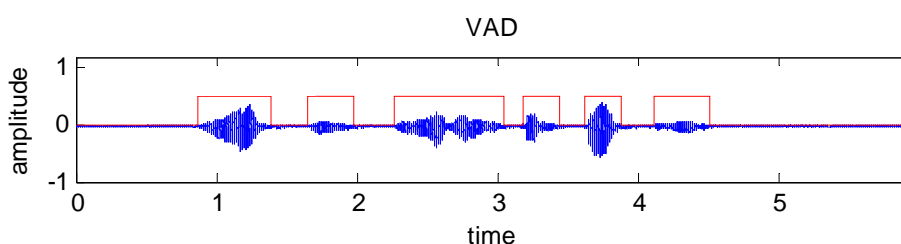


圖 2.1 VAD 的判別結果

用 VAD 來判斷語音，紅線 HIGH 為語音區間

1.4 客觀語音評估

1.4.1 Perceptual evaluation of speech quality (PESQ)

Perceptual evaluation of speech quality (PESQ) 是一個 ITU-T 推薦的客觀語音評估方法，此方法主要考量功率分布(power disturbance)與非對稱(asymmetrical disturbance) 的情況來給予評分，功率分布對語音的影響較大，評分方式則是將這兩個因素量化，再依影響程度大小以線性組合的方式給予一個總合結果的數值。

$$\text{PESQ}=4.5-0.1D-0.031A. \quad (1.5)$$

其中 D 為功率分布的量化數值，A 為非對稱的量化數值，影響程度則由主觀實驗得來

PESQ 最高分數為 4.5，2 以下的時，辨識度與語音品質都頗為糟糕，通常 2.5 以上就有不錯的辨識度，3 以上則是有不錯的語音品質。



1.4.2 SNR segment

SNRseg 則跟一般計算 SNR 的方式無異，但比較的對象只有語音區間，因為想量化語音區間在語音辨識度上的好壞，一般而言語音的能量較雜訊來的大，會有較好的辨識度，陽明方面(語音補償團隊)希望語音在處理過後有 10dB 以上的效果。

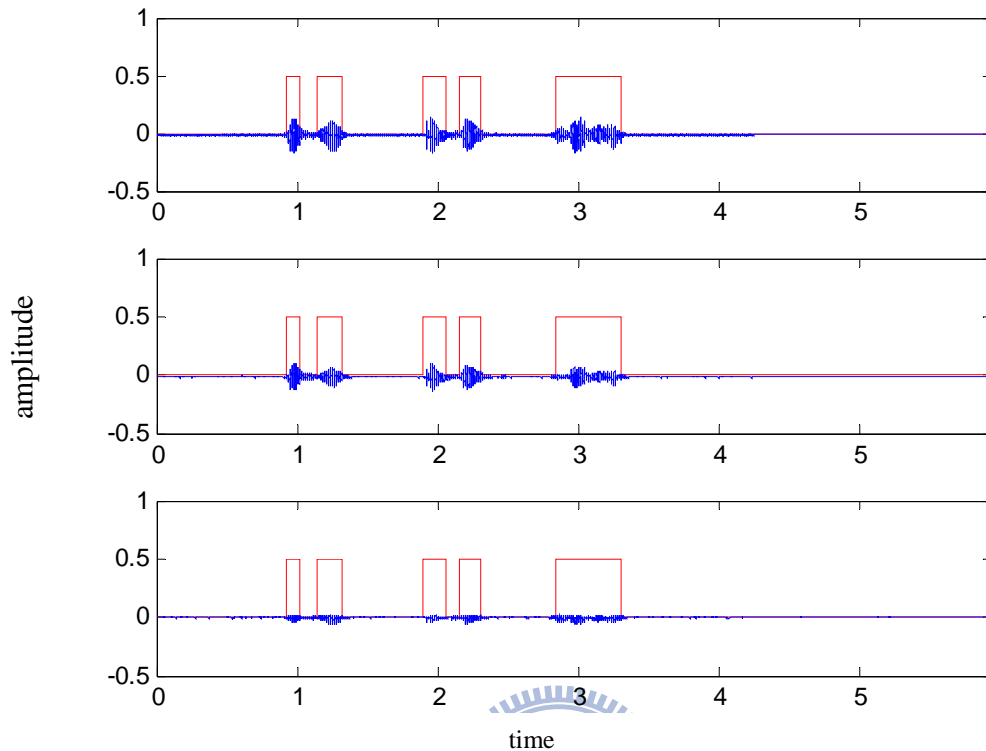


圖 2.2 SNRseg 的計算過程

圖(a)乾淨語音，圖(b)汙染語音，圖(c)雜訊，SNRseg 是利用圖(a)與圖(c)的結果，來計算語音區間的 SNR。

語音比較模式

ITU-T 推薦的 PESQ 比較對象為乾淨語音與演算法處理過後語音的差異，比較範圍則包含語音區間與非語音區間，但若想聚焦在語音辨識度上，則理當比較語音區間即可，因為想確實比較演算在增加辨識度方面提升了多少，但非語音區間並沒有辨識度的問題可言，若一起比較會混淆演算法在語音區間辨識度方面的表現，故在使用 PESQ 方面，有兩種模式，一種是只比較語音區間，另一種則是語音區間與非語音區間皆比較。

1.5 章節概要

第一章 緒論：說明本篇論文的研究動機及前人相關研究。

第二章 消除雜訊演算法的相關工作：主要介紹一些語音的客觀評估方法。

第三章 演算法：包含子空間演算, 頻譜相減演算, 卡爾曼演算及雙耳運算。

第四章 頻譜相減演算應用至濾波器組：包含語音資料庫說明、實驗結果與相關討論。

第五章 卡爾曼演算應用至濾波器組：對研究的結果歸納結論，及提出未來可再改進的方向。

第六章 結論與展望：對研究的結果歸納結論，及提出未來可再改進的方向。



第二章 演算法

一般演算法的主要缺點是會引起語音失真(speech distortion),殘留雜訊(residual noise),及音樂效應(musical noise)方面等問題,通常壓抑殘留雜訊會使語音失真變嚴重,反之亦然,所以需要去權衡殘留雜訊與語音失真影響的程度,避免某一效應太過突出,來提升語音辨識與舒適度,但無法同時完全壓抑這兩種效應。

2.1 子空間演算

2.1.1 子空間原理介紹

子空間演算的主要想法是將污染語音投影至雜訊子空間(noise subspace)與語音子空間(speech subspace),如果可以確實辦到此想法,那就盡可能壓低污染語音在雜訊子空間的分量,保留污染語音在語音子空間的分量,進而達到消雜訊的效果。

劃分雜訊子空間與語音子空間方法,不外乎對污染語音的共變異矩陣(covariance matrix)進行奇異值分解(singular value decomposition)(SVD)或特徵值分解(eigenvalue decomposition)(EVD),理由是,如果特徵值所對應到量值越大,則表示此特徵值所對應包含的語音成分越多。

$$y = x + n \quad (2.1)$$

將污染語音用 y 表示, x 與 n 分別為語音與雜訊

$$R_x = E\{x.x^T\} = P.R_s.P^T \quad (2.2)$$

此式為語音的共變異矩陣

$$\varepsilon = \hat{x} - x = H.y - x = (H - I).x + H.n = \varepsilon_x + \varepsilon_n \quad (2.3)$$

H 為一個線性算子,藉此運算來與污染語音作用,使之還原到近似語音的狀態,上式的最右邊第一項為語音失真,第二項為殘留雜訊

$$\overline{\varepsilon_x^2} = E[\varepsilon_x^T . \varepsilon_x] \quad (2.4)$$

$$\overline{\varepsilon_n^2} = E[\varepsilon_n^T . \varepsilon_n] \quad (2.5)$$

上式兩式分別為語音失真與殘留雜訊的功率

$$\min_H \overline{\varepsilon_x^2} \quad (2.6)$$

$$\frac{1}{K} \overline{\varepsilon_n^2} \leq \sigma^2 \quad (2.7)$$

$$H_{opt} = R_x (R_x + \mu R_n)^{-1} \quad (2.8)$$

藉由解第(2.6), (2.7)式所給的最佳化問題, 則可以得到最佳線性估計子為一個隨 SNR 改變的變數

$$R_x = U \Delta x U^T \quad (2.9)$$

上式為語音的共變異矩陣, 並對其做特徵值(EVD)分解

$$\Delta n \cong \text{diag}(E(|u_1^T n|^2), E(|u_2^T n|^2), \dots, E(|u_K^T n|^2)) \quad (2.10)$$

此式為雜訊子空間與語音子空間重疊的部分, 其餘雜訊子空間的分量不會投影至此

$$R_n \cong U \Delta n U^T \quad (2.11)$$

雜訊的共變異矩陣近似表示式, 若等號成立則第(3.10)式應該為雜訊共變異矩陣的特徵值

$$H_{opt} \cong U \Delta x (\Delta x + \mu \Delta n)^{-1} U^T \quad (2.12)$$

簡化第(2.8)式, 得到較簡化的最佳線性估計子

雖說子空間演算在語音失真與殘存雜訊已經有最佳化過線性算子, 但在得到此算子的過程中包含有一個反矩陣運算, 見(2.12)式, 所以當語音的取樣頻率過大時, 此反矩陣會變大, 運算量將會是一個負擔, 往後也為希望此方法可以運用在濾波器組上。分頻過後, 有 18 個至 54 個頻帶, 如果不分頻都嫌多, 那在實作上則會有相當的難度, 故此放棄了子空間演算在濾波器組上的應用。

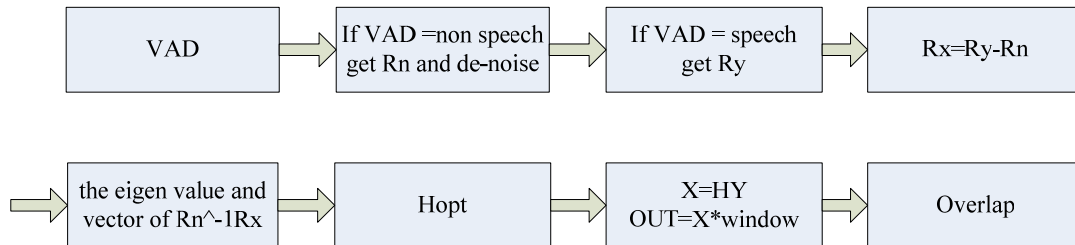


圖 2.1 子空間演算流程圖

Ry 與 Rn 分別為污染語音共變異矩陣和雜訊共變異矩陣, 而 Rx 則為估計出來的語音共變異矩陣

2.1.2 子空間演算語音評比

noise\SNR	-5	0	5	10	15	20
White	2.03	2.46	2.92	3.34	3.58	3.62
Babble	1.61	2.03	2.43	2.82	3.22	3.48
Train	1.37	1.65	2.17	2.56	2.91	3.23
Traffic	1.41	1.80	2.13	2.59	2.95	3.25
Play ground	1.42	1.95	2.04	2.44	2.79	3.09

表 2.1 子空間演算 PESQ

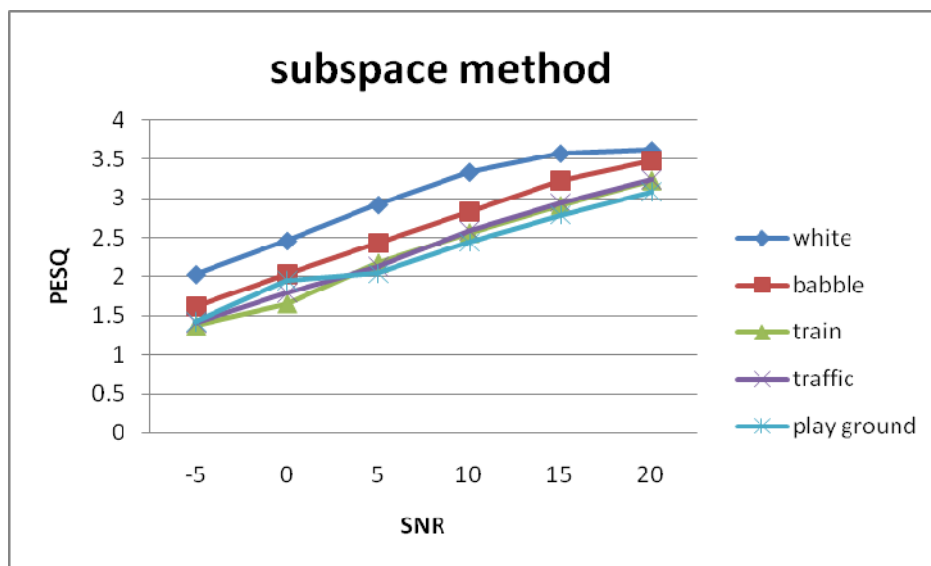


圖 2.2 子空間演算 PESQ

noise\SNR	-5	0	5	10	15	20
White	-16.95	2.99	6.49	16.51	30.40	37.56
Babble	-7.97	6.99	4.18	16.02	25.89	36.18
Train	3.51	13.21	22.08	29.21	33.93	36.44
Traffic	8.34	14.96	22.22	29.04	33.87	36.53
Play ground	-1.21	9.82	19.73	29.57	33.07	36.18

表 2.2 子空間演算 SNRseg

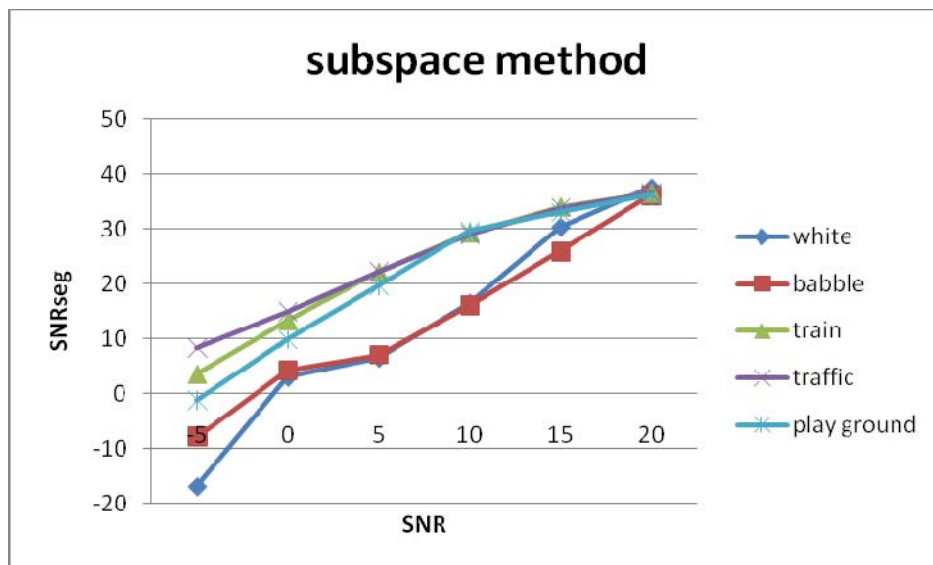


圖 2.3 子空間演算 SNRseg

2.2 頻譜相減演算



2.2.1 頻譜相減演算原理介紹

頻譜相減演算的想法是，將訊號轉至到頻域上，汙染語音在某一頻率上的分量大小為語音與雜訊分量的總合，如果可以確切知道 語音與雜訊在頻率上的分量為何，則可以將汙染語音在某一頻率上的分量，還原至原來語音在某頻率的分量大小，進而達到消雜訊的目的。

通常頻譜相減演算將訊號轉至到頻域，所使用的工具都是 DFT 跟 IDFT，先把 DFT 視為是個分頻的工具，頻譜相減演算的目的是在某頻帶上將雜訊語音分量還原成語音分量大小，也就是只能藉由調整大小這一個變因來進行還原，所以如果頻帶能越平坦則效果理當越好。

順道一提，DFT 的點數多寡，會影響到演算法的好壞，如果 DFT 點數越多，則表示分頻的頻寬會越窄，在頻寬內 PSD 有大幅變化的機會也會較小，會比較平坦，而效果自然也

會比較好。

音樂效應一直是頻譜相減演算的致命傷，造成此效應的原因主要來自頻譜估計，消雜訊的過程中，會將訊號先轉至頻域上，然後再對頻域上分量調整大小，短時間內，調整分量大小範圍比較劇烈(以 short time DFT 來估計雜訊 PSD)，若不夠平順，就很容易出現音樂效應。

$$y(n) = x(n) + w(n) \quad (2.13)$$

將汙染語音用 y 表示， x 與 w 分別為語音與雜訊， n 則表示為時間

$$R_{x,N}(f, i) = \frac{1}{N} |F\{x(i)\}|^2 \quad (2.14)$$

定義 PSD 的計算方式， N 為 DFT 轉換的點數， f 為所對應的頻率， f 的實際大小則取決於取樣頻率， i 則表示為時間區塊編號(time block index)

$$R_y(f, i) = R_x(f, i) + R_w(f, i) \quad (2.15)$$

估計汙染語音的 PSD

$$G_N(f, i) = \frac{R_y(f, i) - k \cdot R_w(f, i)}{R_y(f, i)} \quad (2.16)$$

決定增益函數(gain function)，由語音區間與非語音區間提供汙染語音與雜訊的 PSD，扣除雜訊在某頻帶上對汙染語音占有的分量，然後依照此頻帶帶有多少語音成分，來對汙染語音頻域上的分量做比例調整， K 為一個經驗值

$$\hat{X}_N(f, i) = G_N(f, i) Y_N(f, i) \quad (2.17)$$

用增益函數來調整汙染語音在頻域上的分量大小， Y 為對汙染語音做 DFT

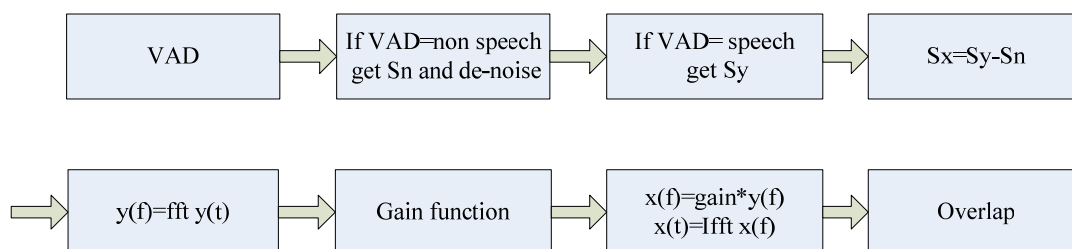


圖 2.4 頻譜相減演算流程圖

S_y 與 S_n 分別為汙染語音與雜訊的 PSD， S_x 則為估計出來的語音 PSD，增益函數為一組調整頻率分量大小的係數

2.2.2 子空間演算語音評比

noise\SNR	-5	0	5	10	15	20
White	1.47	2.06	2.46	2.85	3.14	3.21
Babble	1.47	1.84	2.29	2.64	3.07	3.26
Train	1.05	1.68	2.09	2.54	2.95	3.19
Traffic	1.42	1.80	2.27	2.60	2.94	3.18
Play ground	1.52	1.16	1.87	2.36	2.75	2.98

表 2.3 頻譜相減演算 PESQ

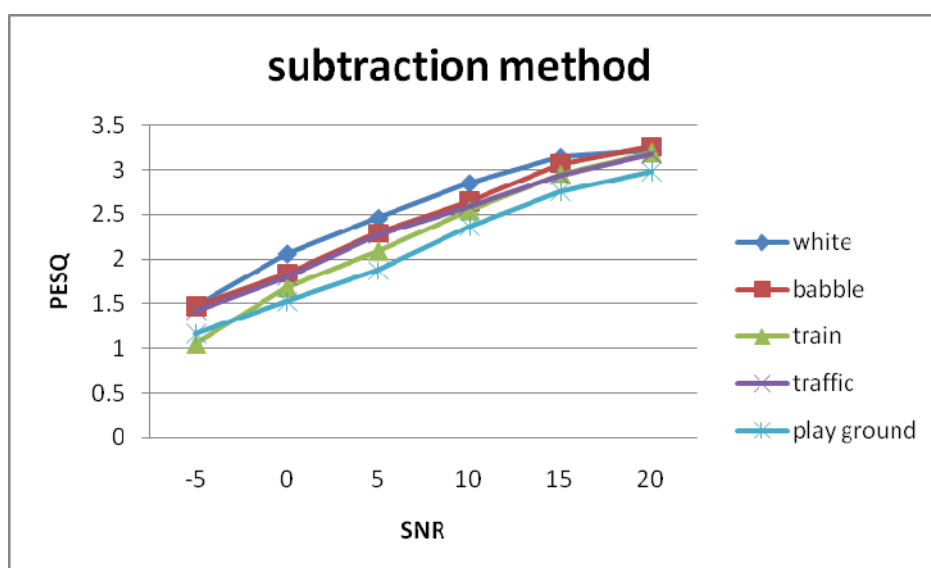


圖 2.5 頻譜相減演算 PESQ

noise\SNR	-5	0	5	10	15	20
White	-23.12	1.13	3.27	18.97	22.70	35.11
Babble	-7.97	4.18	6.99	16.02	25.89	36.18
Train	1.83	12.76	21.22	28.26	33.94	36.43
Traffic	5.02	8.91	17.35	25.88	32.64	36.42
Play ground	-1.19	8.06	16.12	25.21	33.26	36.17

表 2.4 頻譜相減演算 SNRseg

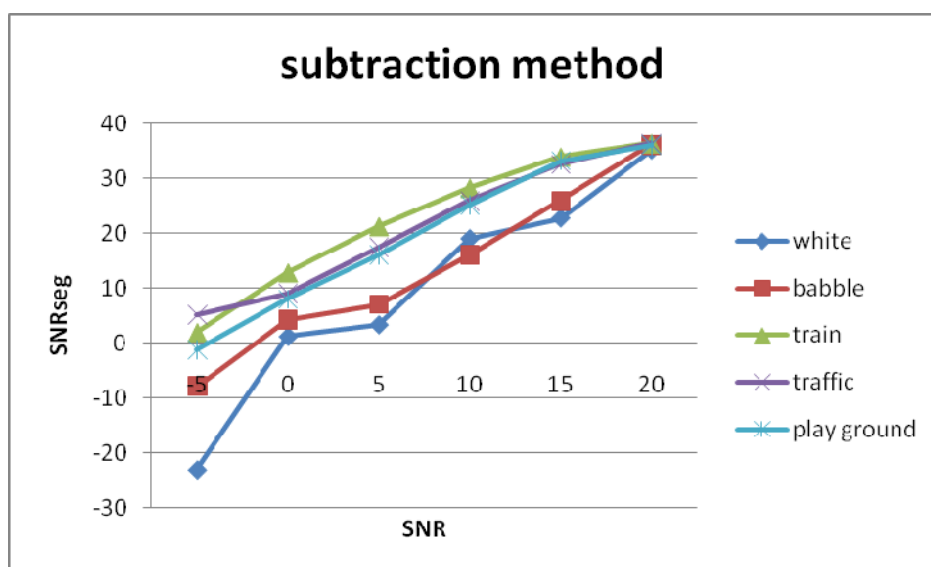


圖 2.6 頻譜相減演算 SNRseg

2.3 卡爾曼演算

在語音處理上，卡爾曼演算是個具有相當成效的一個方法，大部分適應性訊號的書籍都會提及此方法，所以不難想像卡爾曼演算被應用的範圍與程度，此法一般都會使用自迴歸過程 autoregressive process (AR)來描述一段訊號，通常 AR 係數的階數越高效果也會越好，但是伴隨而來的缺點，則是在實作上會面臨到因計算量較大而不易實踐的問題，所以應用此法需在成效與計算量上取得一平衡。

2.3.1 卡爾曼演算原理介紹

自迴歸過程是一種以過去資料為基底，利用其穩態(stationary)特性來預測下一刻資料為

何的一種演算法，這裡的穩態是指現下的資料與過去資料呈現一種穩定的線性關係，所以預測的方式就是將過去資料做一些線性組合，而過去資料所乘上的係數，則稱為 AR 係數，表示方式如下

$$x(n) = a\hat{x}(n-1) + b\hat{x}(n-2) + c\hat{x}(n-3) + w \quad (2.18)$$

X 為目標資料(target data)，x hat 為過去資料，a,b 與 c 為 AR 係數，而 w 為目標資料與預測資料的差異量

卡爾曼演算的想法，就是利用現下語音與過去語音具有穩態特性，選用有效的資料為基底，來預測下一刻的語音為何，進而對污染語音進行還原。

不同於自迴歸過程的地方是基底的選取，自迴歸過程所選用的基底為過去資料，由(2.18)式可知，如果卡爾曼演算也是如法炮製，那麼預測出來的，應該是下一刻的污染語音，而非下一刻的語音，因為基底都是污染語音的集合，再怎麼線性組合，應該還是污染語音。

有鑑於此，卡爾曼演算所選用的基底，是經過修正的，基本上基底的主要元素還是以估出的語音為主，但會依照污染語音與估出語音的差異量，做出一些修正，在得到有效基底後，就可以如自迴歸過程般，利用語音穩態特性來預測下一刻的語音，其表示式如下

$$\hat{x}(n) = \bar{x}(n) + G(n)[z(n) - H\bar{x}(n)] \quad (2.19)$$

X mean 為估出語音，z 為污染語音，G 為增益，會隨污染語音與 x mean 的差力量大小而改變，x hat 則為修正過的基底

$$\bar{x}(n+1) = F\hat{x}(n) \quad (2.20)$$

其中 F 為 AR 係數向量(coefficients vector)，x hat 為基底，x mean 為估出的訊號

接著介紹一個度量語音與估出語音差異的方法，用自迴歸過程估出的語音，主要是參考過去語音經線性組合而來，雖說會很靠近真實語音，但兩這之間還是存在著差異，如果

這差異能越小，則表示估出的語音越準，其運算如下

$$M(n) = E\{[x(n) - \bar{x}(n)][x(n) - \bar{x}(n)]^T\} \quad (2.21)$$

X 為語音，而 x mean 為估出語音

將(2.19) ,(2.20)式代入(2.21)式，計算其存在極小值的情況，就可以確定增益會以何種數學形式呈現，增益的功用是依照語音與估出語音的差異大小來調整基底，期望調整過後的基底經線性組合後，能與真實語音相當近似

$$G(n) = M(n)H^T[HM(n)H^T + R]^{-1} \quad (2.22)$$

H 為[1 0.....0]向量，而 R 為雜訊的變異數(variance)

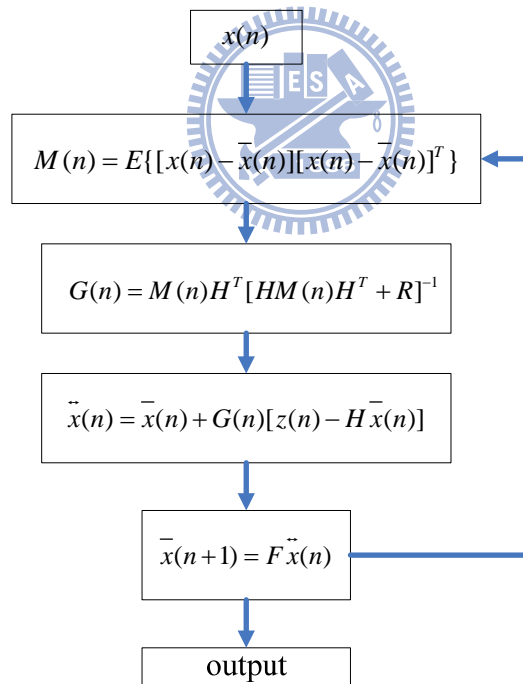


圖 2.7 卡爾曼演算流程圖

對照數學式(2.19)~(2.22)，來進行卡爾曼演算流程

2.3.2 卡爾曼演算語音評比

noise\SNR	-5	0	5	10	15	20
White	1.99	2.25	2.52	2.77	2.95	3.33
Babble	1.86	2.13	2.48	2.90	3.41	3.90
Train	1.87	2.04	2.42	2.75	3.14	3.57
Traffic	1.83	1.97	2.30	2.64	3.14	3.72
Play	0.90	1.82	2.33	2.55	3.00	3.43
ground						

表 2.5 卡爾曼演算 PESQ

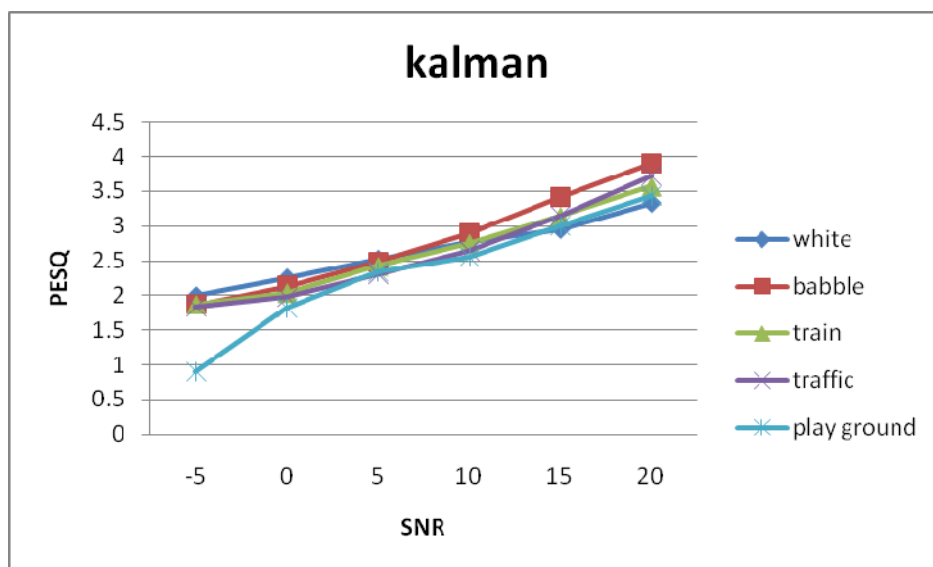


圖 2.8 卡爾曼演算 PESQ

	-5	0	5	10	15	20
White	12.28	19.58	26.64	31.72	32.75	33.08
Babble	2.76	12.59	20.59	28.28	32.25	32.97
Train	5.06	12.15	18.81	25.70	31.63	32.77
Traffic	3.98	12.42	20.04	27.05	32.41	33.08
Play	6.81	13.15	19.33	25.31	32.27	33.05
ground						

表 2.6 卡爾曼演算 SNRseg

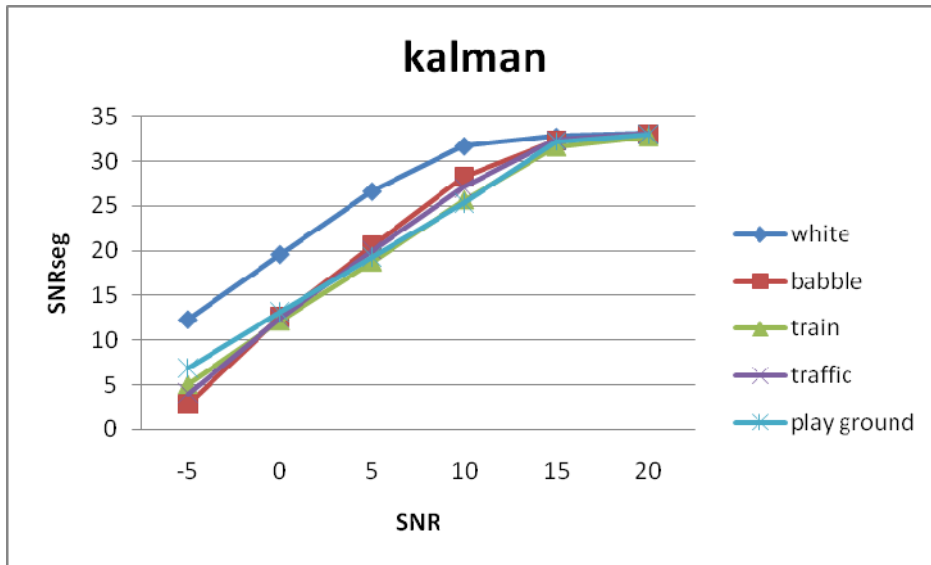


圖 2.9 卡爾曼演算 SNRseg

以下的圖表為演算法的比較。

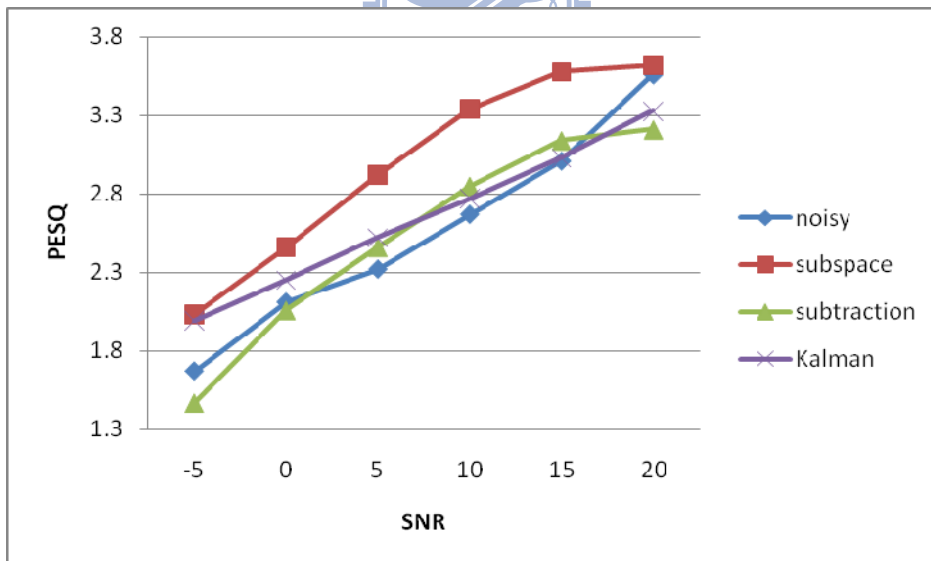


圖 2.10 演算法的比較 1(白雜訊)

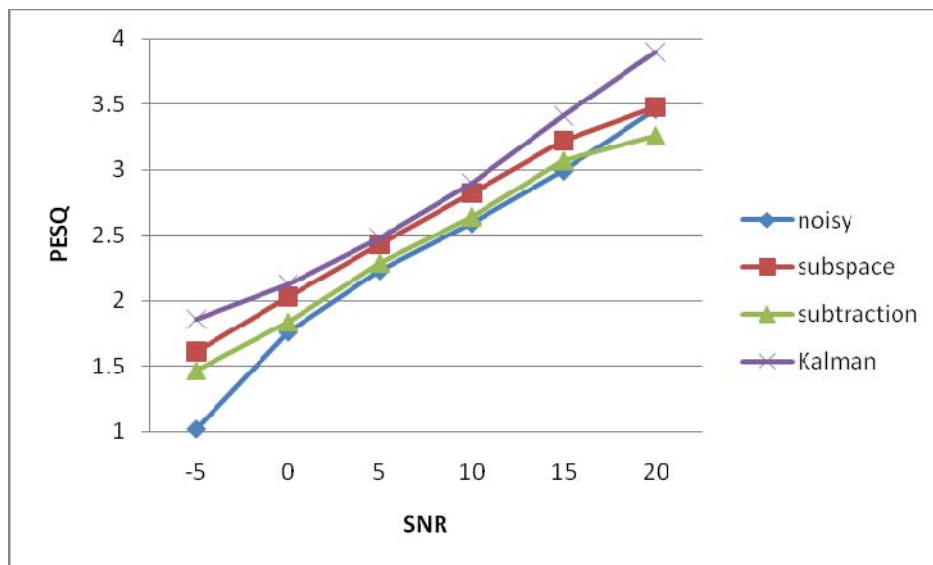


圖 2.11 演算法的比較 2(吵雜人聲)

2.4 雙耳演算

2.4.1 雙耳演算原理介紹



雙耳助聽器演算法，在 90 年代就已經有人提出這樣的想法，在瀏覽今年相關論文的過程中，陸續發現依舊有人在深入研究，但跟很多技術一樣，提出想法到實際生活應用跟量產會有一段時間的落差，就像從布林代數,電晶體到邏輯閘，所以以下會介紹雙耳演算法的假設，實作上會遭遇的問題，以及此法的所帶來的好處跟優勢。

雙耳演算法最重要的想法，就是雜訊到達左右耳的時間不同，因為左右耳皆有個別接收器，而人跟人對話過程，大多距離不會太遠或是講話者通常在正前方，所以接收到的語音視為同時到達左右耳，以上是雙耳演算模型的基本假設，以下有附圖說明。

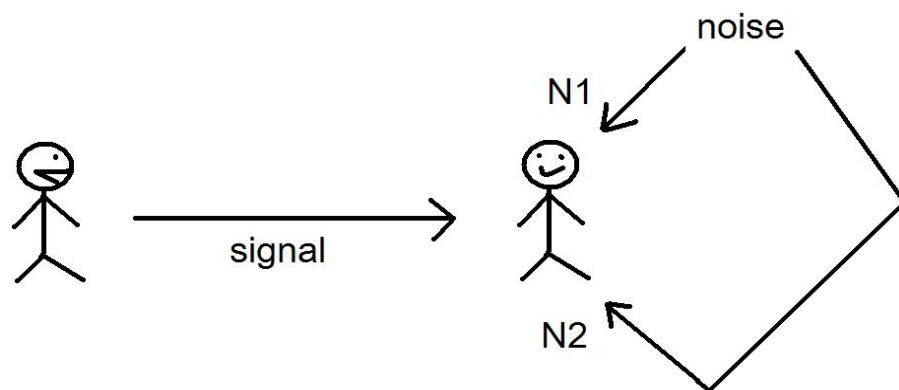


圖 2.10 雙耳演算模型

signal 為乾淨語音，N1 與 N2 皆為雜訊，只是到達左右耳的時間不同

當然還有許多其他因素如:語音是否也有時間差，雜訊和語音到達左右耳是否強度一樣，雜訊源若為兩個以上等問題，現階段暫不考量，先以理想狀況來處理。

由上圖可以看出，雜訊到左右耳的時間不同，所以可以分別寫出下列左右耳接收到語音的情況

$$y_1 = s(t) + n(t) \quad (2.23)$$

$$y_2 = s(t) + n(t - \tau) \quad (2.24)$$

第式為右耳 第式為左耳 $s(t)$ 為語音 $n(t)$ 為雜訊 τ 則為雜訊到達左右耳時間差

有了上述兩式左右耳的數學式，發現若將其相減，將可以得到純雜訊的訊息，有了較充分的雜訊訊息，對消雜訊而言將會是一項利多

$$u = y_1 - y_2 = n(t) - n(t - \tau) \quad (2.25)$$

τ 為雜訊到達的時間差

當然將左右耳接收訊號在時域上相減，比較難看出其用處，但經 DFT 運算，可以看到有獲得較完整雜訊 PSD 的機會，但還需要一些處理，表示式如下

$$U(w) = N(w)(1 - e^{-i w \tau}) \quad (2.26)$$

U 為 u 經 DFT 運算 N(w) 為雜訊 PSD τ 則為雜訊到達的時間差

接下來同通訊零消(zero forcing)的想法，在得到 U 的情況下，希望能還原到 N 的狀態，而由式，也可以看出 H 為何，以下則將左右耳所需的 H 一次表出，因左右耳有一個到達時間差，所以在頻域上 H 會有一個 $e^{-i w \tau}$ 的差異

$$N(w) = U(w)H(w) \quad (2.27)$$

同 zero forcing 尋找一個 H 來使 U 還原至 N

$$H_1(w) = \frac{1}{(1 - e^{-i w \tau})} \quad (2.28)$$

$$H_2(w) = \frac{e^{-i w \tau}}{(1 - e^{-i w \tau})} \quad (2.29)$$

H1 H2 分別為左右耳所需的 H



現在只剩下 τ 時間差的計算，尋找時間差的方法是利用訊號在時間上的相關性，u 所包含的成分只有兩個幾乎一樣的雜訊相減，但這兩個雜訊之間具有一個時間差，而左右耳收到訊號的成分則包含語音跟雜訊，訊號與訊號自行本身內積運算會有極大值，所以把 u 與 y1 做內積運算，再將 y1 位移一段時間，重複剛才的內積運算，當找到最大值的時侯，就可以知道 τ 時間差為何了，其想法寫成數學式如下

$$\tau = \arg \max_t [(n(t) - n(t - \tau)) * y(t)] \quad (2.30)$$

利用訊號在時間上的相關性 來尋找 τ 時間差 n 為雜訊 y 為右耳接收到的訊號

如此一來在得知較完整雜訊 PSD 方面，算是告一段落，接著是在實作上所遭遇的問題，左右耳在接收到訊號後，需要做資料傳輸或是資料交換的動作，所以會面臨到的問題有傳輸率，傳輸方式為無線或有線，何種傳輸媒介以及消耗功率等，相較一

般演算法這些會是雙耳演算法額外所需的代價。

雙耳演算法所帶來的好處有兩點，第一 提供較完整的雜訊 PSD 訊息，第二 不論在語音區間或非語音區間，皆可以估計雜訊，這是單耳演算一個相當大的限制，就是只能在非語音區間估計雜訊，而估計雜訊的精準與否，往往對演算法的效果有絕大的影響。

以下的附圖是說明雙耳演算的過程，direction finding 主要在尋找雜訊到達左右耳的時間差，compensator 則是類似零消，由時間差來計算出最佳的 H，能夠從 U 來得到雜訊 PSD，黑色虛框部分則是頻譜相減演算，只是相較以前的單耳演算法，現在擁有更精準的雜訊估計，可以提升頻譜相減演算的效果。

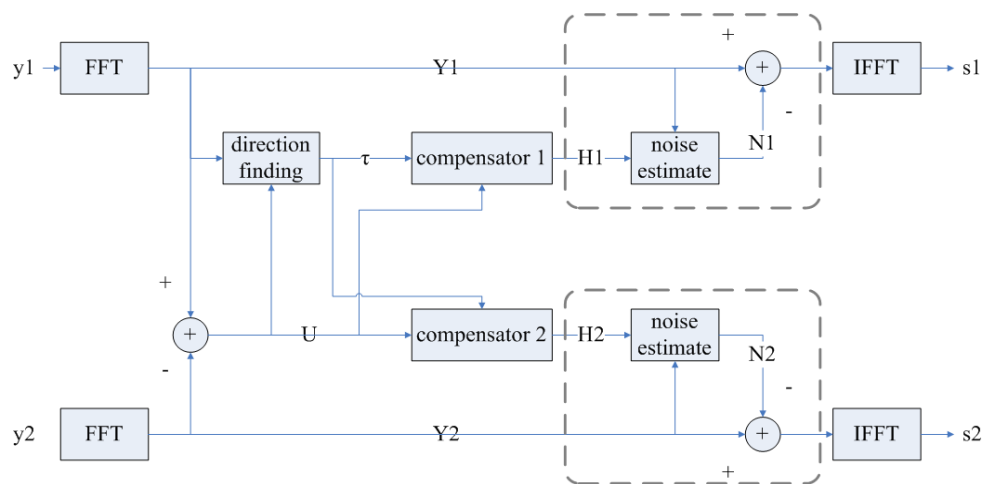


圖 2.11 雙耳演算流程圖

y_1 與 y_2 為左右耳接收到的污染語音， τ 為雜訊到達左右耳的時間差， U 為 y_1 與 y_2 做 DFT 後相減， H_1 與 H_2 是用來使 U 還原成雜訊 N_1 和 N_2 的運算子，黑框部分效果則雷同頻譜相減演算， s_1 與 s_2 則為處理後語音

優勢	1.提供更完整的雜訊 PSD 資訊 (可以加強頻譜相減演算或其他演算的效果) 2.估計雜訊不受 VAD 限制
代價	1.計算量變大 且運算時間變久 2.需計算雜訊的時間差 以及 H_1, H_2 3.左右耳需進行資料交換所涉及的相關問題

表 2.7 雙耳演算的優勢與代價



第三章 濾波器組與頻譜相減演算應用

3.1 濾波器組

3.1.1 濾波器組設計規格

人類語音大部分的訊息量，都集中在較低頻的部分，可聽的頻率範圍為 20~20000Hz，根據硬體的規範，語音的取樣頻率為 24k，將其切成 18 個頻帶，來符合人類語音特性，所以設計上高頻帶的頻寬較寬，低頻帶較窄，然後可以針對患者是在那個頻帶聽力受損，給予語音補償。

在結構方面，為了使硬體可以共用，將 18 band 分成六組，每組含有 3 個濾波器，第一組濾波器先從高頻區塊開始，第二組濾波器的產生就由第一組濾波器減少取樣(down sample)來得到，第三組濾波器也是由第二組減少取樣得來，之後濾波器的產生就往下依此類推即可，其結構圖如下

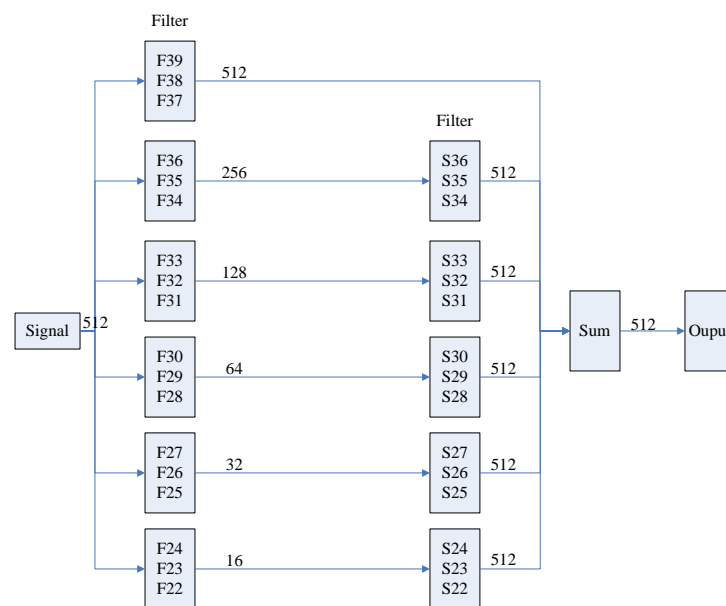


圖 3.1 濾波器組結構圖

由圖所示，濾波器組包含分頻濾波器(Ana filter)與合成濾波器(syn filter)，在兩者之間可以對患者做語音補償，分頻濾波器部分，第一組與第二組的差異有以下兩點，一 因為減少取樣之故，所以第二組的語音資料量為第一組的一半。二 第二組濾波器的頻寬為第一組頻寬的一半，第三組的資料量與頻寬皆為第一組的四分之一，往後依此類推。合成濾波器部分，則是將語音資料增加取樣(up sample)過後，使每一頻帶的資料量相同，再合成輸出。

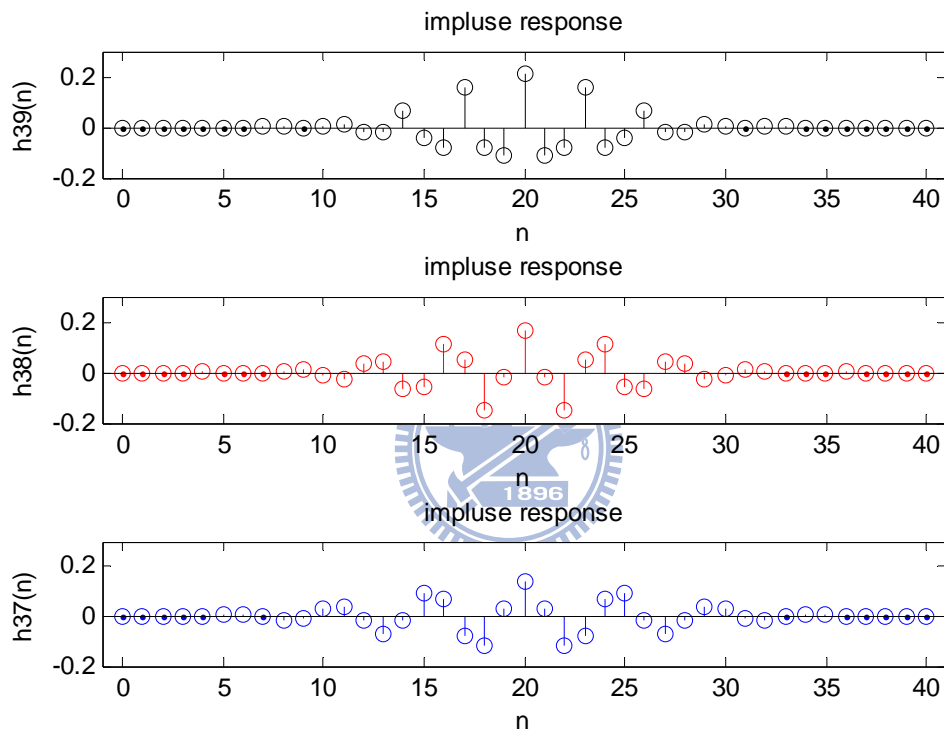


圖 3.2 濾波器的脈衝響應
以上為濾波器 F39 F38 及 F37 在時間上的脈衝響應

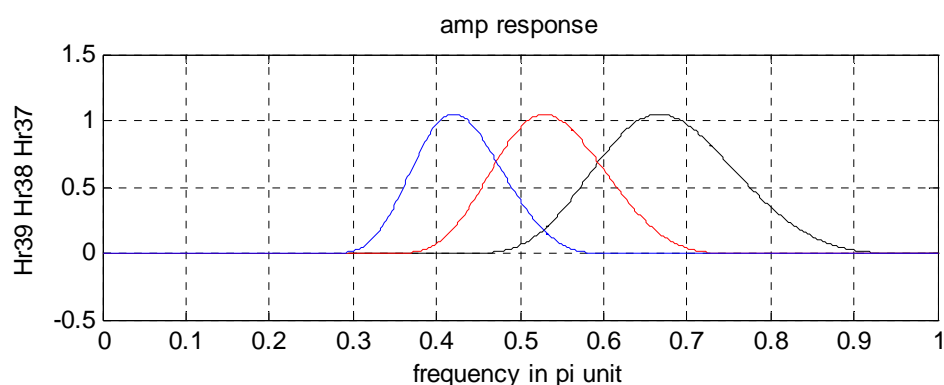


圖 3.3 濾波器的頻率響應

以上為濾波器 F39 F38 及 F37 在頻率上的頻率響應

Band index	Freq range	BW	Band index	Freq range	BW
Hr 39	7200Hz~8880Hz	1680Hz	Hr 38	5680Hz~7200Hz	1580Hz
Hr 37	4520Hz~5680Hz	1160Hz	Hr 36	3600Hz~4440Hz	840 Hz
Hr 35	2840Hz~3600Hz	790 Hz	Hr 34	2260Hz~2840Hz	580 Hz
Hr 33	1800Hz~2220Hz	420 Hz	Hr 32	1420Hz~1800Hz	395 Hz
Hr 31	1130Hz~1420Hz	290 Hz	Hr 30	900Hz ~1110Hz	210 Hz
Hr 29	710Hz ~900Hz	196 Hz	Hr 28	565Hz ~710Hz	145 Hz
Hr 27	450Hz ~555Hz	105 Hz	Hr 26	355Hz ~450Hz	98Hz
Hr 25	282Hz ~355Hz	72Hz	Hr 24	225Hz ~277Hz	52Hz
Hr 23	176Hz ~225Hz	49Hz	Hr 22	141Hz ~178Hz	36Hz

表 3.1 波器組的頻率規範

取樣頻率為 24k Hz，以 magnitude 為 0.707 當頻寬的邊界

3.1.2 濾波器組的副作用

濾波器組雖然在結構上有共用的優點，可以節省硬體，但也會產生時間延遲的缺點，第二組濾波器的產生，是由第一組濾波器減少取樣得來的，所以也就是要等第一組先跑過一次，才會產生第二組，也就是 series 的概念，所以在得到第六組訊號的同時，也產生了不少時間延遲。

第一組的濾波器所用的 tap 數為 41，減少取樣的動作也需要 34 個 tap 數，所以分頻濾波器部分的時間延遲為 $20 \times 32 + 17 \times 31 = 1167$ taps。合成濾波器部分則是 $20 \times 31 = 620$ taps。取樣頻率為 24k，所以造成約略 75ms 的時間延遲。

3.1.3 54 頻帶濾波器組設計

54 band 較 18 band 而言有較高的頻率解析能力，對頻譜相減演算應用在濾波器組有相當的幫助，如之前所述，頻譜相減演算只能藉調整頻譜分量大小，來進行消雜訊，所以頻帶能越平坦則效果會越好，而相較於 18 band，54 band 的頻帶會較窄，語音訊號在頻帶內 PSD 的變化，也比較不至於太過劇烈，會較適合頻譜相減演算與卡爾曼演算的使用。

現有規格將 24kHz 切成 18 band，依 ANSI 標準，照患者聽力受損的頻帶給予增益來補償，但 54 band 有較高的頻率解析，所以為同時滿足這兩個條件，在設計上還是以 18 band 的架構為基礎來做修改，以下用以頻譜相減演算應用至濾波器組為例來做說明

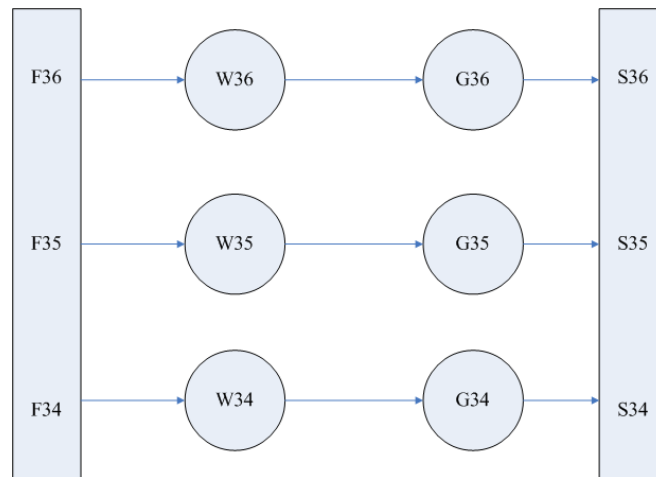


圖 3.4 18 頻帶濾波器組

F36~F34 為分頻濾波器，S36~S34 為合成濾波器，而 G36~G34 為依 ANSI 標準給予患者的增益補償，W36~W34 則是頻譜相減演算用來調整頻帶分量大小的權重，以上為 18 band 的濾波器組（請與圖 4.1 濾波器組結構圖對照）

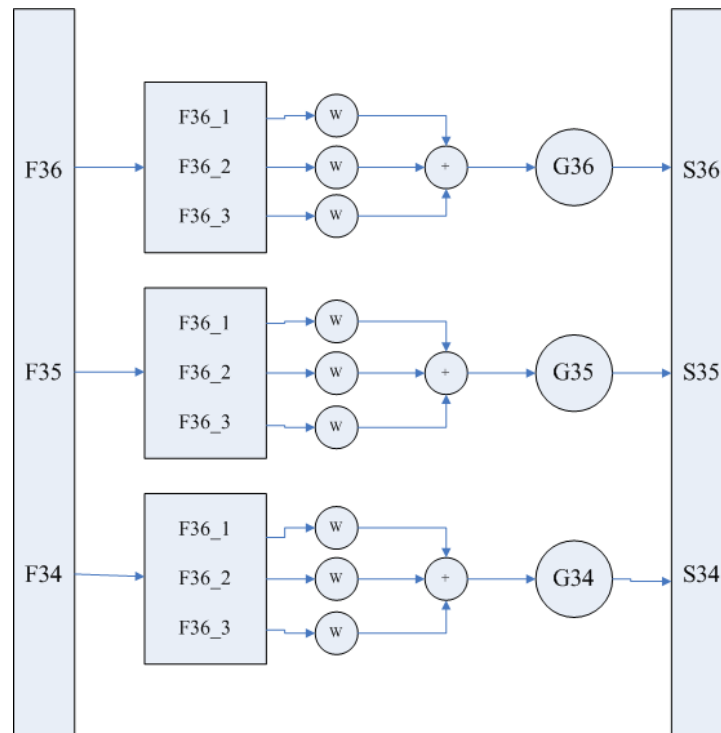


圖 3.5 54 頻帶濾波器組

在 F36~F34 分頻濾波器，S36~S34 合成濾波器，而 G36~G34 為增益補償方面與圖，完全一樣，唯一不同在於每個頻帶又會再細分為三個子頻帶，這樣一來在使用頻譜相減演算上會有較多調整權重的空間，以上為 54 band 的濾波器組

由上述可知，只需將原先的 18 band 濾波器組再經過窄頻的濾波器，便可得到 54 band 濾波器組，但為合乎 ANSI 標準，訊號經過三個子頻帶的濾波效果應該要等價於一個頻帶，以強度響應(magnitude response)來看，子頻帶相加後能相當近似於之前的頻帶，在帶頻(pass band)方面應盡可能維持平坦，每個子頻帶在強度大約 0.707 的位置重疊，來使交接點能維持平坦，而細分的子頻帶重要性皆相當，因頻寬大攜帶訊息多也會較多，所以子頻帶的頻寬也都一樣，大約為原先頻帶的三分之一左右。

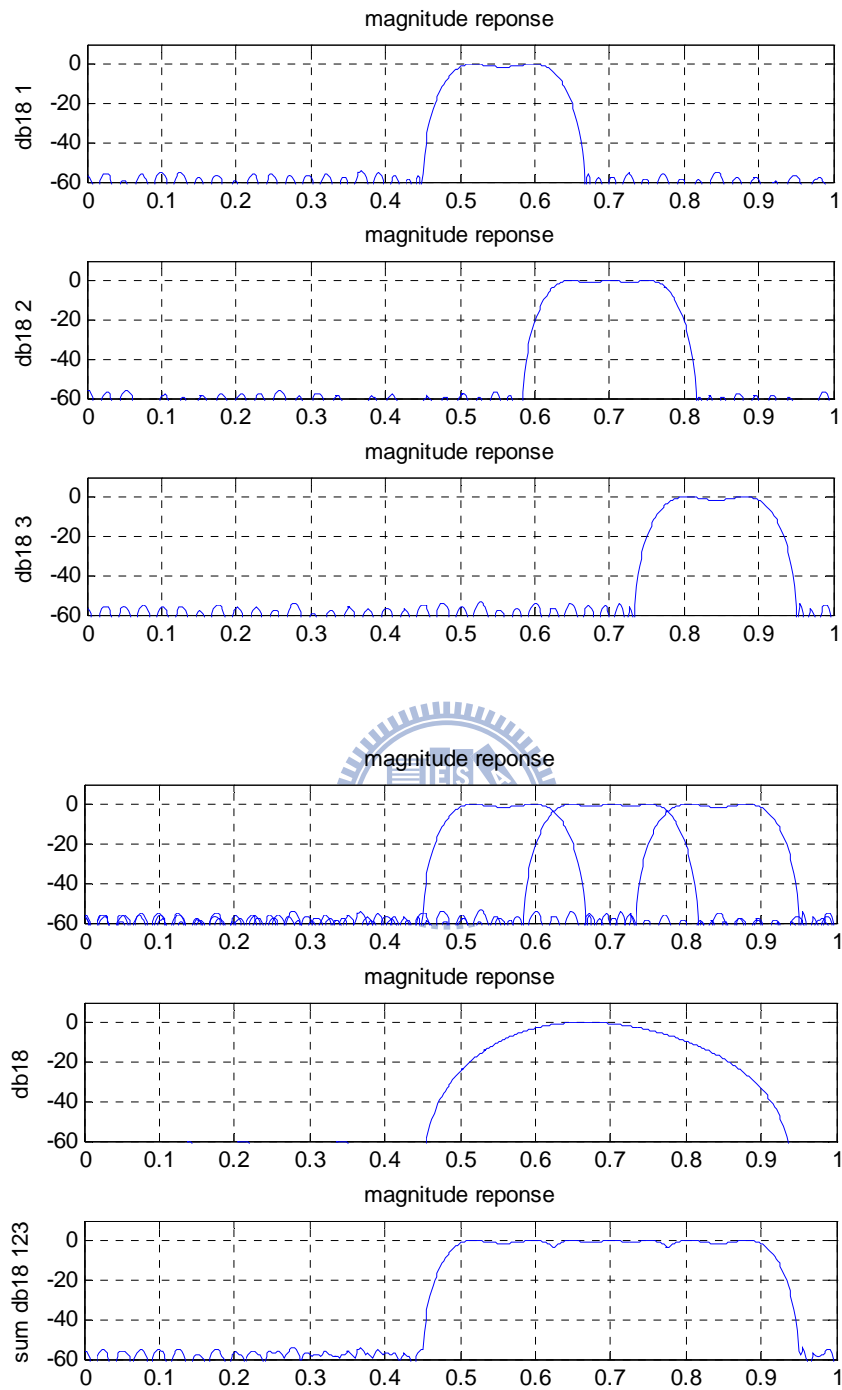


圖 3.6 18 頻帶與 54 頻帶的頻率響應

一個 18 band 的頻帶可由三個 54 band 的子頻帶來組成 (a)(b)(c)為子頻帶的頻率響應，
(d)三個子頻帶重疊的位置，(e)頻帶的頻率響應，(f)三個子頻帶等價的頻率響應

3.2 頻譜相減演算應用

3.2.1 頻譜相減演算的分頻設計

頻譜相減演算藉由 DFT 運算，將訊號從時域轉至頻域，訊號會分佈在等區間上頻率軸上，然後去調整這些訊號的頻域分量大小，來降低雜訊。

濾波器組本身具有分頻的功能，所以將頻譜相減演算放置到濾波器組上，可以由濾波器組本身的特性來取代 DFT 運算，較為不同的地方是，訊號不會從時域轉至頻域，時域訊號經過濾波器還是在時域上，訊號不會等區間分佈在頻域軸上，為符合規範，濾波器在設計上，高頻頻帶寬，低頻窄。

在調整頻域分量大小，來降低雜訊的精神，則是不變，若 DFT 一次作用的點數有 64 點，則可視作分頻為 64 個頻帶，而濾波器組只有 18 個，頻譜相減演算調整頻率分量大小的多寡，是依據訊號 DFT 後在頻率軸(頻帶)上含有多少能量，來決定那頻帶能保留多少比例。



至於計算能量方式，則大同小異，頻譜相減演算只需看訊號經 DFT 後在頻率軸(頻帶)上的分量多寡即可，在濾波器組上，因 Parseval 定理，將分頻過後的時間訊號平方再相加一段時間後，即可看出頻帶上所含的能量多寡。

3.2.2 頻譜相減演算應用至濾波器組

如之前所述，頻譜相減演算應用在濾波器組上，由分頻濾波器取代 DFT，合成濾波器取代 IDFT，在調整分量大小上，頻譜相減演算對頻域訊號作用，濾波器組則是對時域訊號調整，為說明比較其差異，其流程與結構如下

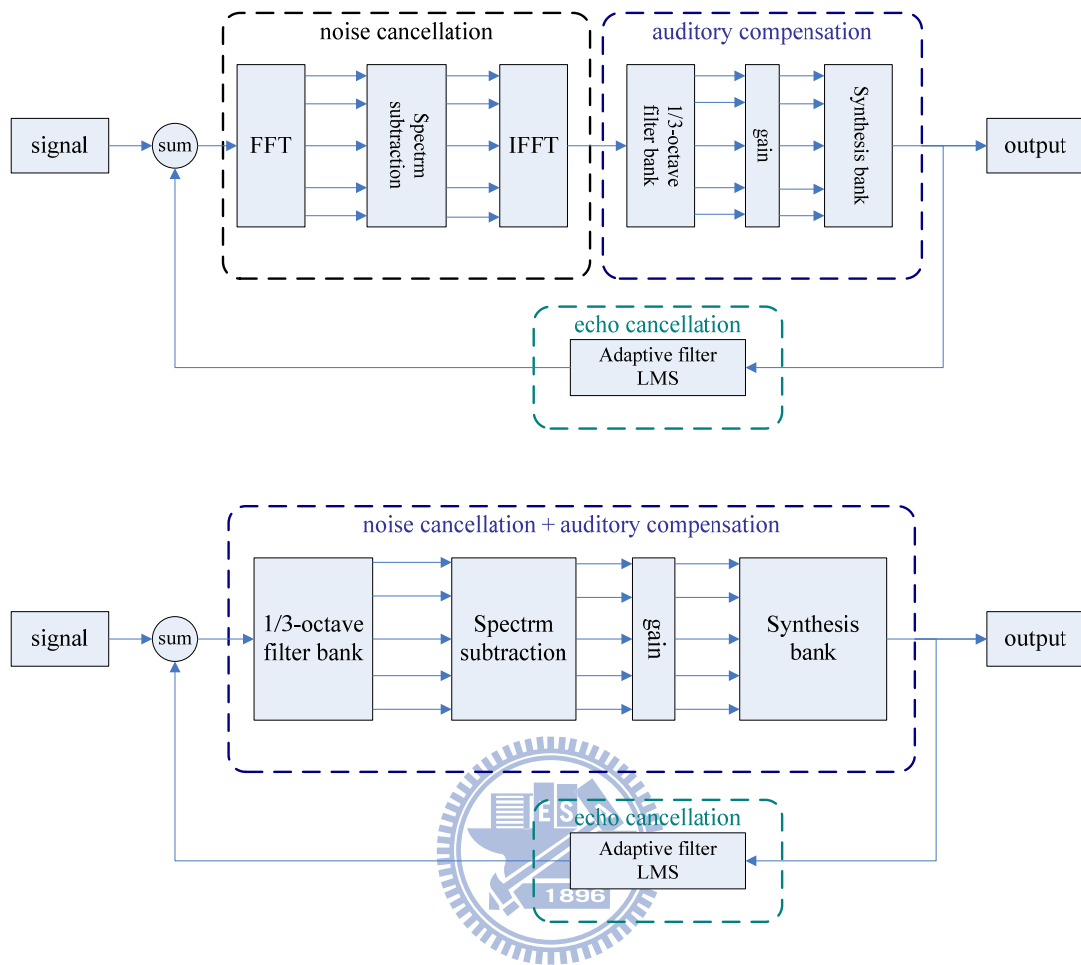


圖 3.7 頻譜相減演算應用至濾波器

圖(a)直接將頻譜相減演算套用至濾波器組，圖(b)共用濾波器組的分頻功能，取代 DFT 運算

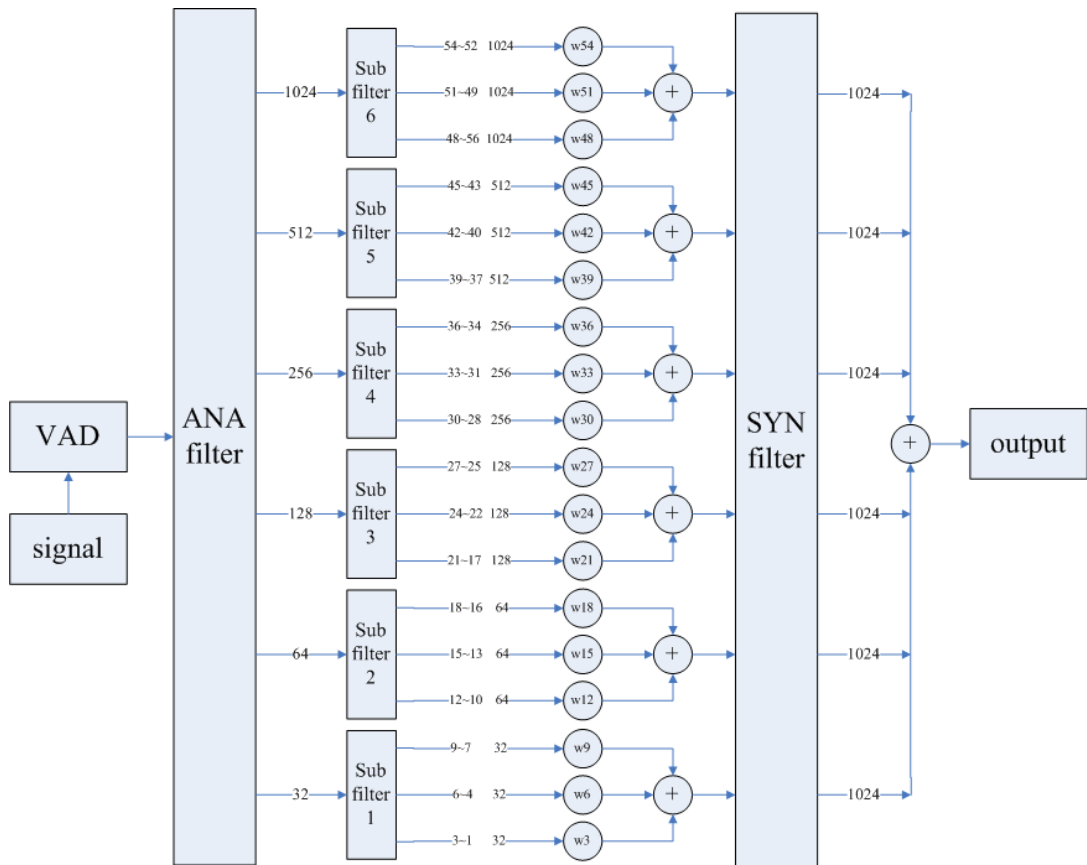
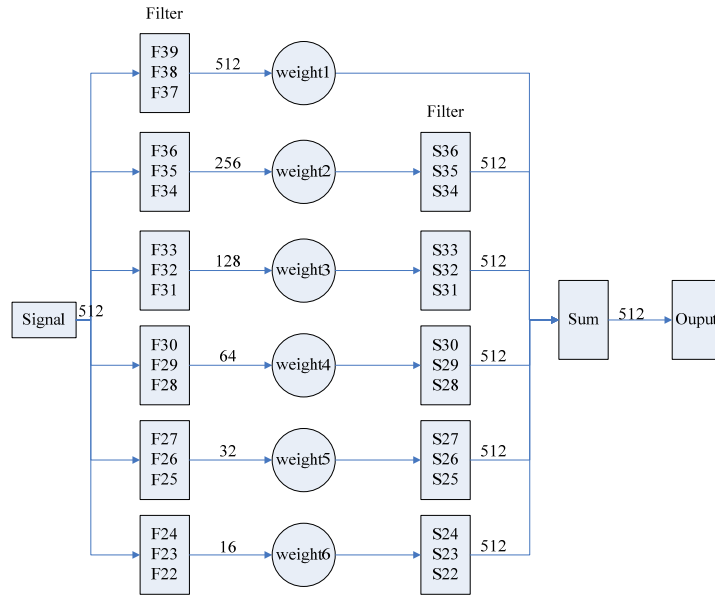


圖 3.8 頻譜相減演算與濾波器組結構圖

(a)權重為增益函數，用來調整頻帶上訊號大小(b)54 band 濾波器組

對頻譜相減演算而言，若 DFT 一次作用的點數有 64 點，就有 64 個權重可拿來調整分量大小，濾波器組則只有 18 個權重可用，相較之下在頻率解析方面略嫌不足，所以將濾波器組拆至 54 個濾波器，期待能對頻率分量做出更細微的調整。

3.2.3 權重更新方法

訊號在經過演算法處理過後，往往會出現“音樂效應”，頻譜相減演算也不例外，而音樂效應會出現的主因是，當在估計現下汙染語音(current noisy speech)與現下雜訊(current noise)頻譜時，若與過去頻譜落差大，便會使權重在更新時的變化太過劇烈，就會出現音樂效應。

改善音樂效應的方法，原則上就是維持權重在時間上的連續性，也就是盡量避免權重劇烈變化的情況發生，一般來說估測現下頻譜的時間拉長，權重則不會在短時間內有太大變化，但估測時間的拉長也會導致，無法立即反應背景雜訊的變化，所以在更新的過程中，最好能具有在短時間內反應，且變化幅度不會過大的特性，除非現下汙染語音與現下雜訊改變甚大。

在保持權重連續性方面，採指數衰減的方式來更新，此方法在更新權重的過程中會以某種比例保留過去權重，而非完全採用現下權重，來減少權重變化過大的情況發生，其流程如下

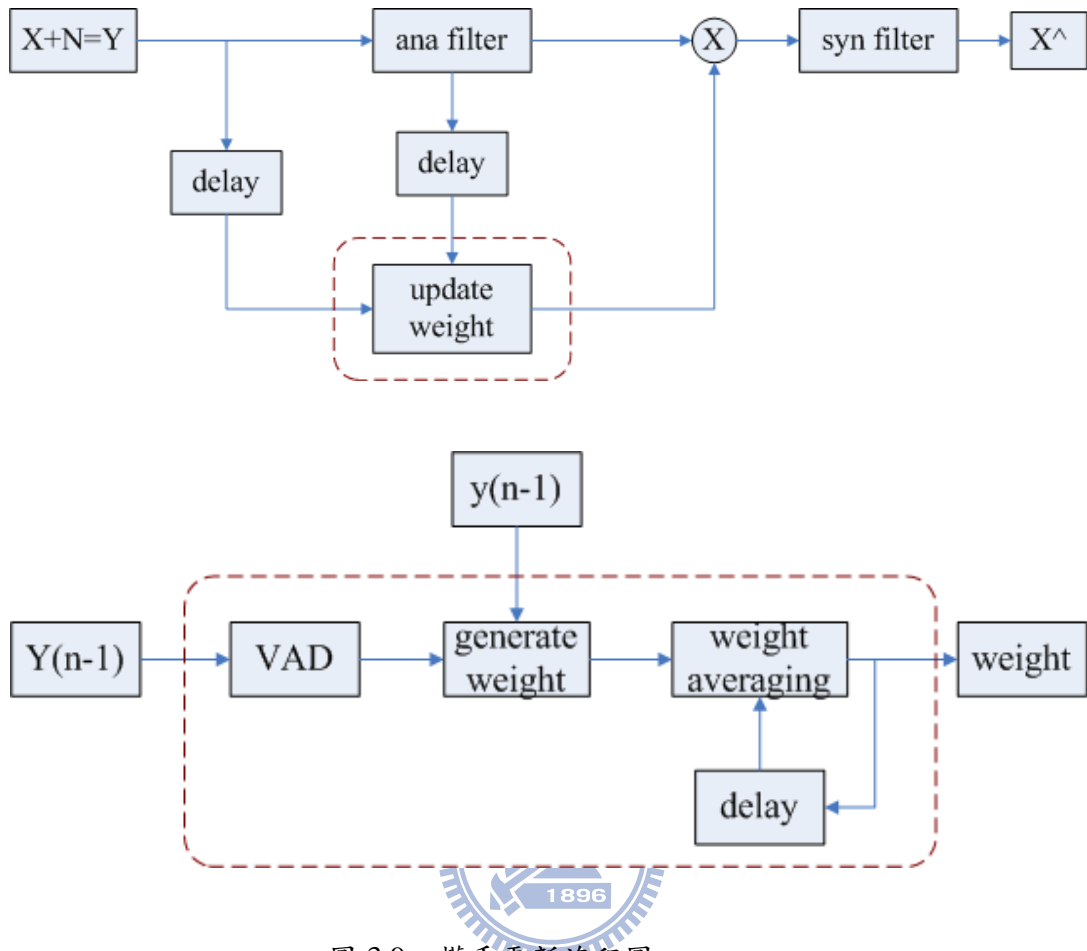


圖 3.9 權重更新流程圖

圖(a)分頻濾波器與合成濾波器對訊號分別有分頻跟合成的作用，虛線框部分則是有緩和權重變化的功用，圖(b)虛線框內的較細部的說明，Y(n-1)為未分頻的訊號，主要是用來判定是否為語音區間，使演算法在非語音區間更新雜訊資訊，在語音區間更新污染語音資訊，y(n-1)則為分頻訊號，用來決定現下權重，而 weight averaging 則是決定要參考多少過去權重，來使權重變化不至於太過劇烈

$$w_i(n) = \frac{p_y(i) - 0.7 \times p_n(i)}{p_y(i)} \quad (3.1)$$

以上為決定現下權重的方式，Py 與 Pn 分別為頻帶上的功率，i 為頻帶的編號，n 則表示為時間

$$\bar{w}_i(n) = a(n) \times \bar{w}_i(n-1) + (1-a(n)) \times w_i(n) \quad (3.2)$$

$a(n)$ 為保留比例， $w(n-1)$ 為過去權重， $w(n)$ 為現下權重，而 w mean 為 $w(n-1)$ 與 $w(n)$ 線性組合的結果

式中的 $a(n)$ 可以決定現下權重具有多少參考價值，在想法上如果過去權重與現下權重近似，則表示過去權重具有參考價值，可以給予某種比例保留來繼續沿用，若差異太大，則應該適度捨棄，甚至完全不採計。

以下考慮三種更新的情況：

第一類，發生在非語音區間，若估測訊號的 PSD 與過去雜訊 PSD 相近，則可以沿用，不需大幅更新。

第二類，也是發生在非語音區間，但估測訊號的 PSD 與過去雜訊 PSD 有某種程度上的差異時，表示過去雜訊 PSD 的參考價值不高，需大幅更新或不採計。

第三類，發生在語音區間，因為是語音區間，所以估測訊號 PSD 與過去雜訊 PSD 一定會有所差異，所以幾乎一定會更新，更新的幅度則由估測訊號 PSD 與過去雜訊 PSD 的差異大小來決定，而更新幅度的計算方式如下

$$\beta = \min\left\{\frac{\sum |p_y(i) - p_n(i)|}{\sum p_n(i)}, 1\right\} \quad (3.3)$$

P_y 與 P_n 分別為頻帶上污染語音與雜訊的能量， i 為頻帶的編號，而更新幅度的上限是 100%

確定了更新幅度，就可以調整 $a(n)$ 保留比例，來反映過去權重具有多少參考價值，而計算 $a(n)$ 的情況有以下兩種，當 $a(n-1) \leq 1-\beta$ 時，往往更新幅度不大，表示現在沒有大幅調整保留比例的趨勢，過去權重有參考價值，當 $a(n-1) > 1-\beta$ 時，一般來說更新幅度較大，

則表示現下，必須大幅調整保留比例，過去權重的參考價值也較低，保留比例的調整方式如下

$$a(n) = r \times a(n-1) + (1-r) \times (1-\beta) \quad \text{for } a(n-1) \leq 1-\beta \quad (3.4)$$

$$a(n) = (1-\beta) \quad \text{for } a(n-1) > 1-\beta \quad (3.5)$$

$a(n)$ 為保留比例， B 為更新幅度， $r=0.7$ 則為一個經驗值

以下為對語音在兩種不同型態的雜訊，做更新權重來觀察其結果，更新權重目的在於，希望對每段話都有不同的權重，來反映對背景雜訊的變化，但又不希望權重變化過大，以至於有音樂效應的產生，所以預期的結果應該是，權重有所變化，但整體來說每段話與每段話之間差異不會太大

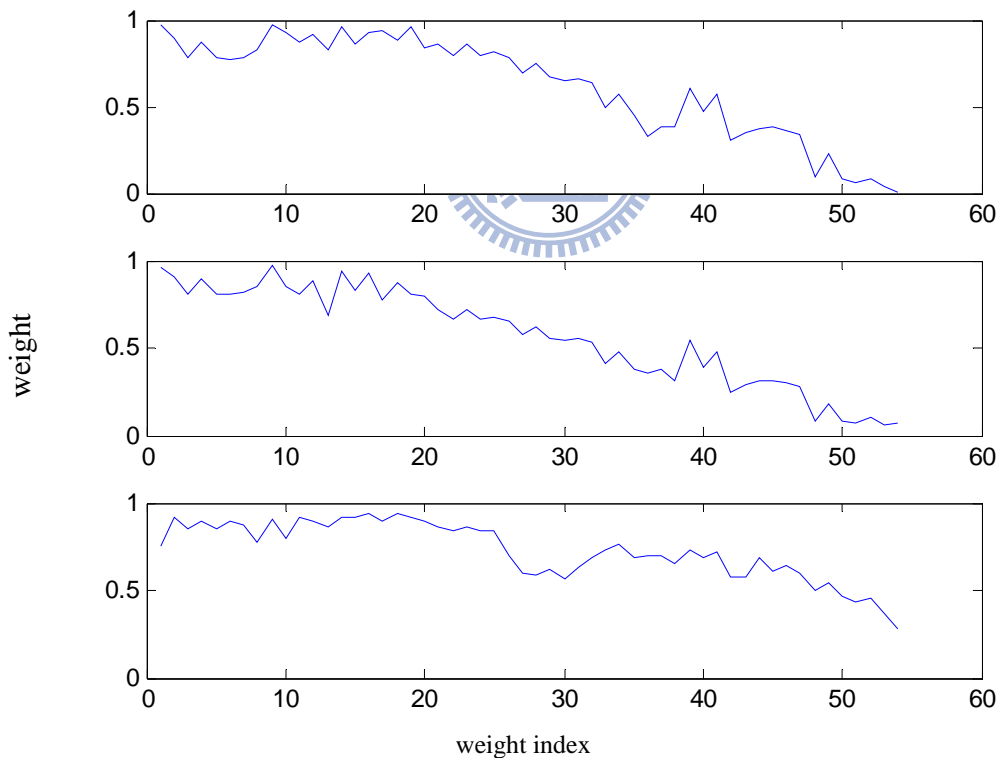


圖 3.10 權重更新 1

圖(a)(b)(c)分別為不同每段話的權重，以白雜訊為背景雜訊，所以權重分佈會比較類似語音本身的 PSD 分佈情況。

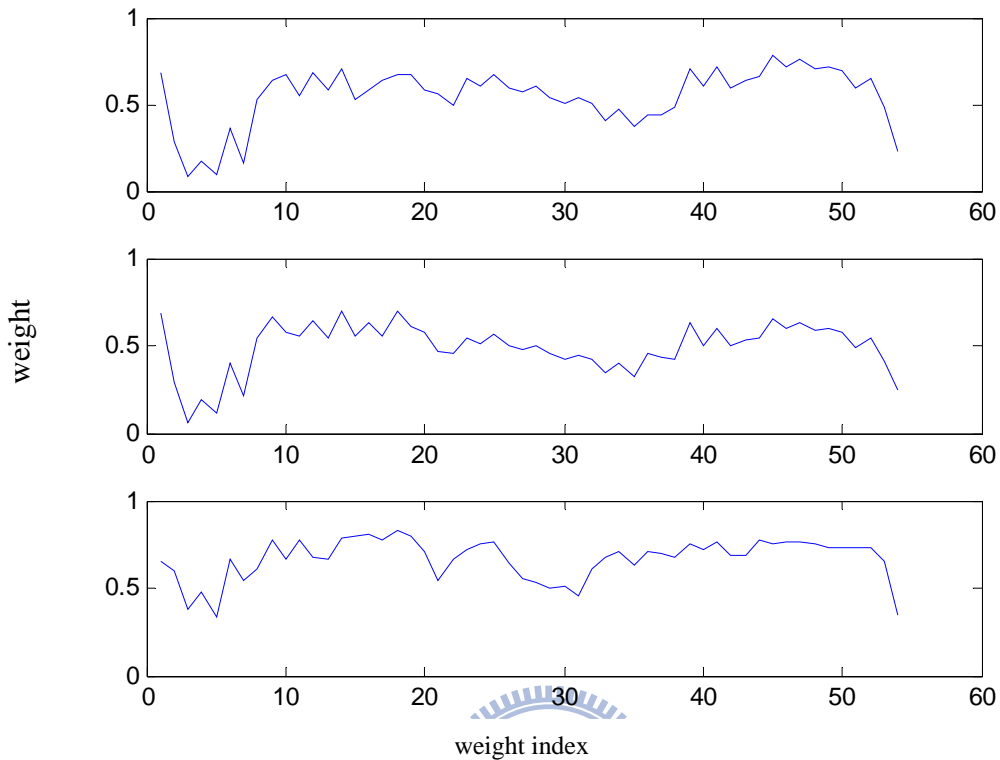


圖 3.11 權重更新 2

圖(a)(b)(c)分別為不同段話的權重，以 babble 為背景雜訊，因為 babble 的低頻成分多，所以給予的壓抑也比多。

3.3 頻譜相減演算應用之語音評比

3.3.1 頻譜相減演算應用至 54 頻帶語音評比

noise\SNR	-5	0	5	10	15	20
White	2.35	2.60	2.86	3.17	3.38	3.48
Babble	1.83	2.16	2.53	2.96	3.31	3.56
Train	2.20	2.19	2.48	2.82	3.17	3.42
Traffic	1.71	2.05	2.36	2.76	3.25	3.56
Play ground	1.97	2.03	2.30	2.67	3.03	3.29

表 3.2 頻譜相減演算 54 頻帶 PESQ

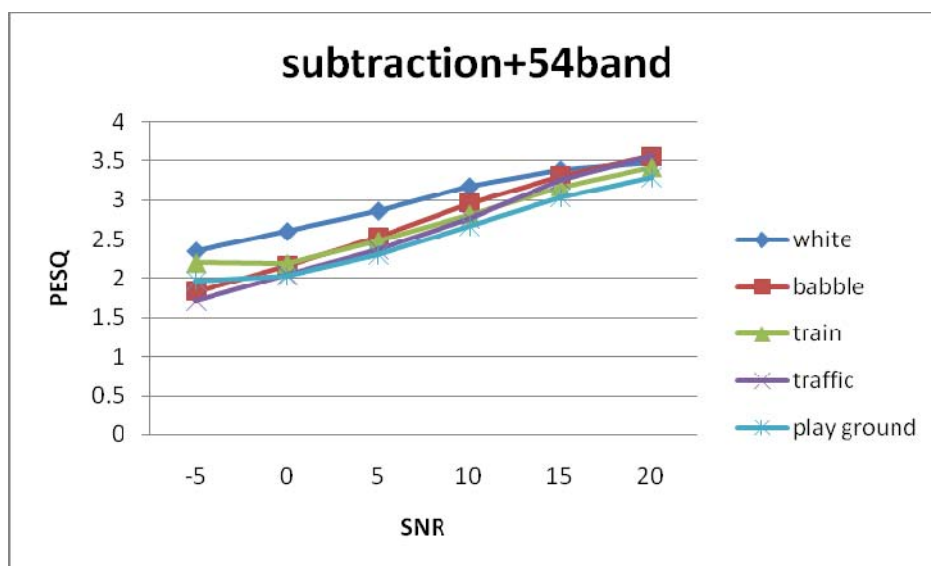


圖 3.12 頻譜相減演算 54 頻帶 PESQ

noise\SNR	-5	0	5	10	15	20
White	18.37	23.95	30.08	33.79	36.15	37.48
Babble	2.80	5.29	14.08	23.26	29.73	33.99
Train	9.50	11.01	18.94	24.75	31.39	35.43
Traffic	3.27	6.81	14.86	23.00	30.08	34.73
Play ground	8.64	13.52	20.68	26.76	31.48	34.88

表 3.3 頻譜相減演算 54 頻帶 SNRseg

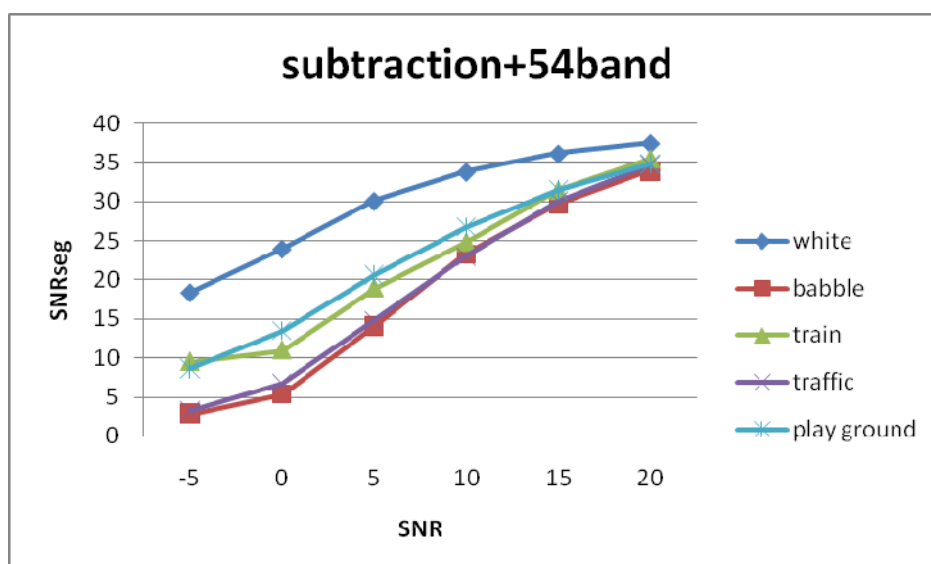


圖 3.13 頻譜相減演算 54 頻帶 SNRseg

3.3.2 頻譜相減演算應用至 18 頻帶語音評比

noise\SNR	-5	0	5	10	15	20
White	1.99	2.09	2.24	2.56	3.02	3.39
Babble	1.69	1.94	2.12	2.36	2.52	2.57
Train	1.63	1.95	2.10	2.31	2.47	2.53
Traffic	1.59	1.75	2.01	2.24	2.39	2.56
Play ground	2.35	1.49	1.94	2.15	2.35	2.44

表 3.4 頻譜相減演算 18 頻帶 PESQ

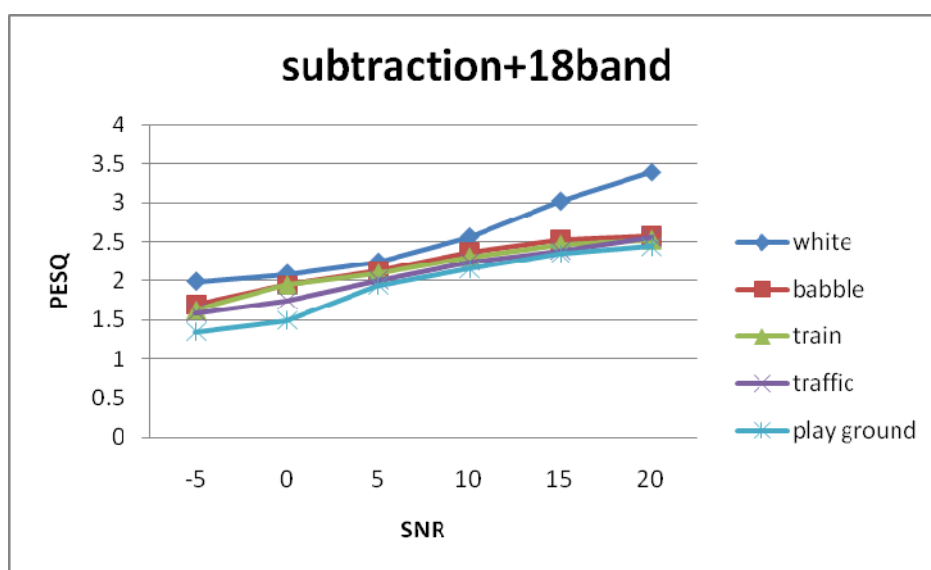


圖 3.13 頻譜相減演算 18 頻帶 PESQ

Noise\SNR	-5	0	5	10	15	20
White	15.99	18.92	22.07	28.78	32.22	36.10
Babble	4.02	11.24	15.78	23.74	30.68	31.91
Train	8.11	12.37	19.20	25.86	31.07	34.35
Traffic	6.77	13.35	20.18	26.39	30.97	32.03
Play ground	12.14	18.06	23.92	28.60	32.09	34.58

表 3.5 頻譜相減演算 18 頻帶 SNRseg

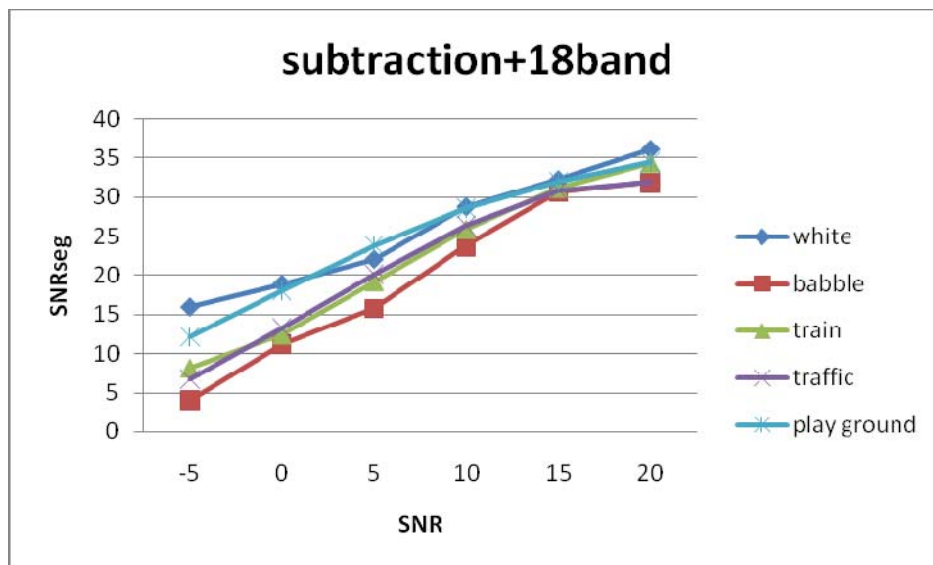


圖 3.14 頻譜相減演算 18 頻帶 SNRseg

在整體方面，如果相鄰頻帶的權重變化過大不夠連續時，會造成語音品質下降，每個頻帶在計算權重都是彼此獨立計算，在對雜訊做 short time DFT 時，其 PSD 變異會較大，所以在權重過於敏銳的情況下，有時反而會發生 PESQ 較低的情況，當然本身對頻率的解析能力過低，則在 PESQ 方面也不會有太好的表現。

而消除殘留雜訊方面，分頻越細，對頻率的解析能力較好，可調整的權重數也多，理當效果也會好。

第四章 濾波器組與卡爾曼演算應用

4.1 卡爾曼演算的分頻設計

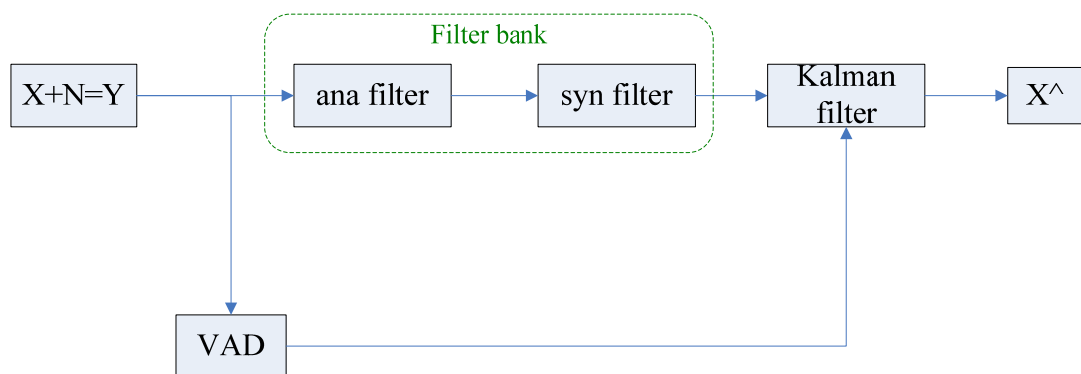
之前所介紹的卡爾曼演算，是以過去語料為基底以及其穩態特性來還原語音，自迴歸過程方面，如果階數越高則表示參考過去越多的語料為基底，就如泰勒展開式一般，越多的多項式基底，會越接近想表的數學式，但也會使泰勒展開式變的龐大計算變的繁雜，而 AR 係數也有這樣的特性。

若數學式本身很單純，例如為一個常數，那麼泰勒也只需要常數項就足夠表達，所以如果可以將語料的組成成分變單純，那麼 AR 係數理當也不需要使用到太高階，就會有不錯的效果。

寬頻的語料可想而知會在組成成分上比較複雜，故需要較高階的 AR 係數來表示，分頻是個將複雜語料變單純的方法，結構較單純的子頻(subband) 語料，只需要低階的 AR 係數即可，而濾波器組本身就有分頻的功能。

分頻帶來的另一個好處是，增加 AR 係數的收斂速度，因為語料結構變得較簡單，相對的子頻 PSD 也會比寬頻 PSD 來得平坦，可以在較短的時間內取得有效的 AR 係數。

此外相較於雜訊變異數的估計，AR 係數的精準度就顯得比較沒那麼嚴苛，因為略為不準的 AR 係數並不會對整個演算法造成太大的傷害，但有偏差的雜訊變異數則對卡爾曼演算的結果影響頗深。



圖

4.1 卡爾曼演算應用流程圖

此為卡爾曼演算流程圖，濾波器組作分頻之用

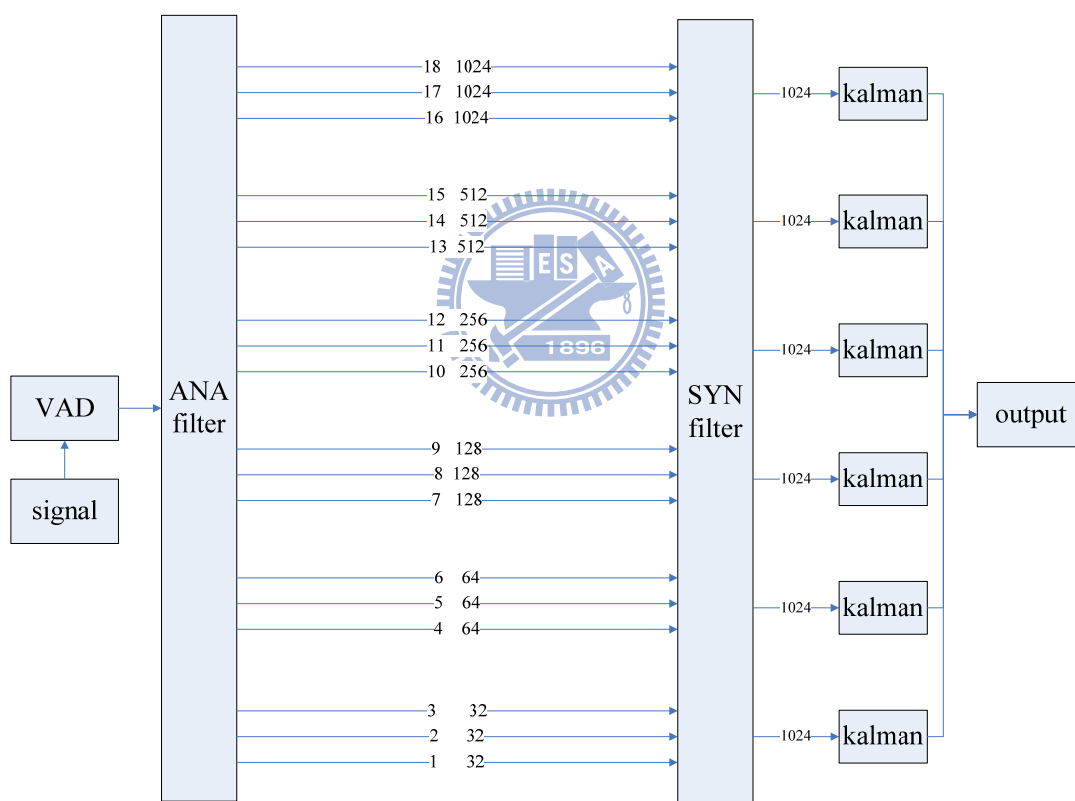



圖 4.2 卡爾曼演算應用結構圖

此為卡爾曼演算結構圖，分頻濾波器與合成濾波器分別用來對訊號作分頻與合成之用，在分頻濾波器與合成濾波器之間的數字，分別為濾波器編號與段落點數，合成濾波器與卡爾曼演算之間的數字，則為訊號經過合成濾波器之後的段落點數。

其實頻譜相減演算與卡爾曼演算的相似之處，在兩者都希望語料單純一點和在 PSD 上看起來平坦一點，前者藉調整頻率分量大小，來使汙染語音 PSD 能接近語音 PSD 因為只能調大小，所以頻帶內的 PSD 越平坦越好，後者的 AR 係數轉至頻域上觀察，如果與語音 PSD 越接近，則表示在時間上能越精準的描述語音穩態的特性，因為使用的是低階 AR 係數，所以頻帶內的 PSD 最好不要有劇烈變化，也是越平坦越好。

卡爾曼演算只應用在 18band 而非 54band 的理由是，54band 對卡爾曼演算來說分頻過細，反而會增加計算量，頻譜相減演算一個段落所需的計算量，大概會較卡爾曼演算一個點多一些，但一個段落的點通常有數百個，所以單位點來看，當然卡爾曼演算計算量會較多，而且 18band 相較於寬頻的情況，頻帶內 PSD 的變化情況，已經平坦很多，甚至為了再減少計算量，還可以將 18band 合成到 6band，可以少約略 2/3 的計算量。

4.2 卡爾曼演算應用至18頻帶語音評比



noise\SNR	-5	0	5	10	15	20
White	2.03	2.26	2.58	2.84	3.18	3.62
Babble	1.79	2.06	2.44	2.94	3.46	3.88
Train	1.78	1.92	2.06	2.76	3.19	3.65
Traffic	1.63	1.83	2.19	2.62	3.21	3.72
Play ground	1.71	1.89	2.08	2.46	3.03	3.44

表 4.1 卡爾曼演算 18 頻帶 PESQ

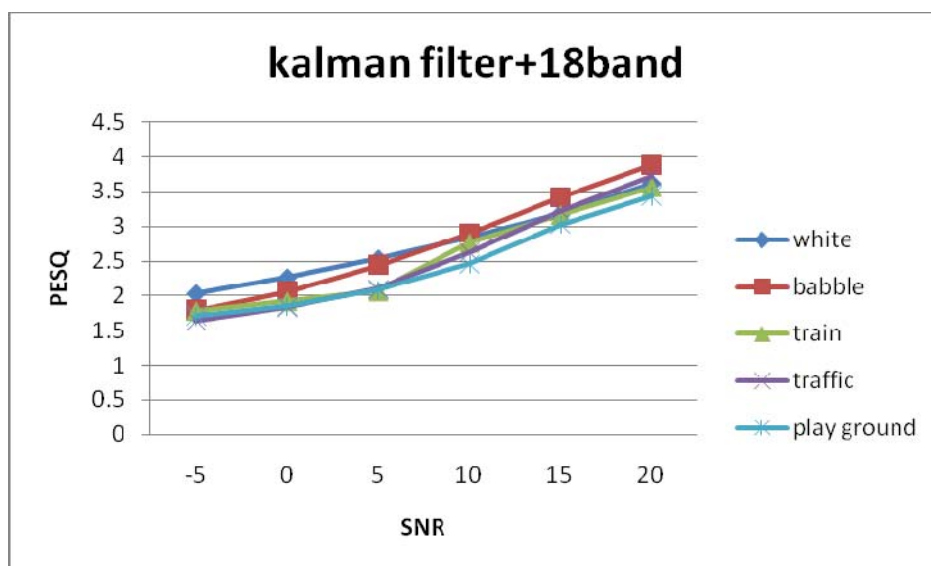


圖 4.3 卡爾曼演算 18 頻帶 PESQ

Noise\SNR	-5	0	5	10	15	20
White	2.04	19.07	28.69	33.53	35.37	37.52
Babble	5.93	12.74	23.98	30.27	32.70	34.15
Train	2.36	11.58	20.17	27.56	30.34	31.52
Traffic	4.68	14.04	22.15	27.38	30.56	33.08
Play ground	6.10	14.86	24.45	30.93	32.93	33.48

表 4.2 卡爾曼演算 18 頻帶 SNRseg

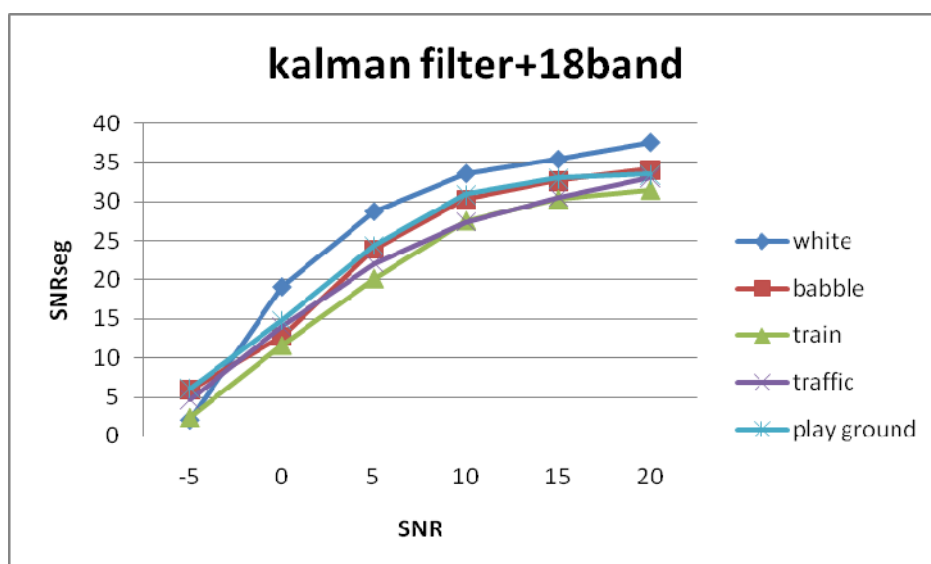


圖 4.4 卡爾曼演算 18 頻帶 SNRseg

以下的圖表為演算法應用在濾波器組的比較。

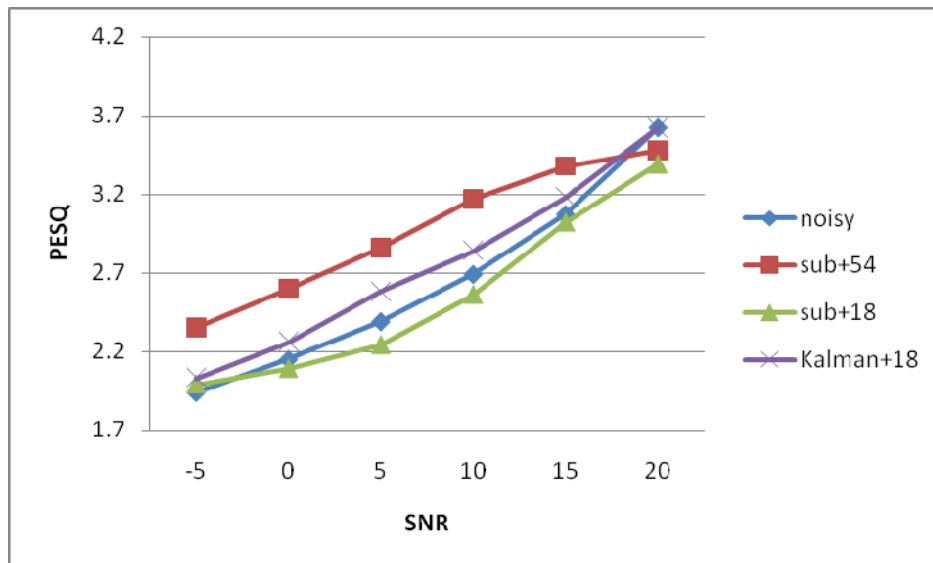


圖 4.5 濾波器組演算比較 1(白雜訊)

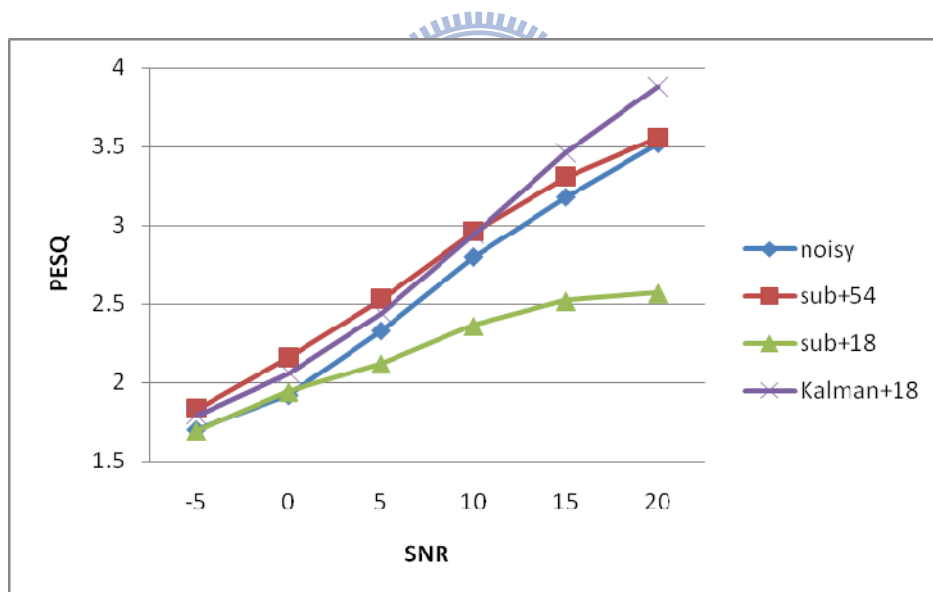


圖 4.6 濾波器組演算比較 2(吵雜人聲)

第五章 結論與展望

5.1 結論

為使病患在受損的頻率能得到補償，濾波器組架構勢必一定會存在，所以在選擇使用演算法上，自然也會挑選能有運算共用或是提升效能機會的演算法，在子空間演算，頻譜相減演算與卡爾曼演算等演算法中，子空間演算比較不受青睞，主因是濾波器組的分頻動作無法與 SVD 或 EVD 結合共用，在效果表現方面也沒有提升。

濾波器組對頻譜相減演算與卡爾曼演算則是有幫助，理由之前已說明，但對頻譜相減演算來說，18band 的分頻不夠細，在表現上與 54band 有段落差，而 54band 又會使濾波器產生比較長的延遲，對實作又可能會是一個問題。

卡爾曼演算在分頻過後，會更適合 AR 模型，在計算量方面當然會增加，但也無需每個頻帶都個別處理，也可將 18band，分成 12band 或是 6band，低頻區塊分細一些，來使運算量與效果達到平衡。

而雙耳演算法的主要功效為，在估計雜訊方面能給予較完整的資訊，且雙耳演算的相容性相當高，所以不論是何種演算法，應該都能因此而提升消雜訊的效果，來使各種演算法有機會可以更接近理論。

5.2 演算法回顧與比較

以下將各個演算法的優缺點以及摘要以表格的方式來做比較

Method	優點	缺點
子空間演算	1.以 PESQ 評比來看，對白雜訊頗為有效。	1.音樂效應 2.有反矩陣計算，會使計算量變大。 3.SVD 或 EVD 在硬體設計上會較 DFT 為複雜
頻譜相減演算	1.想法及運算不會過於複雜。 2.適合在濾波器架構下使用。	1.DFT 與 IDFT 會產生音樂效應 2.權重變化過快也會有音樂效應 e
卡爾曼演算	1.對 AR 係數的精準度，要求不高。 2.適合在濾波器組架構下使用。	1.AR 係數的階數高，才能描述語音 PSD 的情況，會增加計算量。 2.需精準估計雜訊變異數
雙耳演算	不論為語音區間與否，皆可估計雜訊，無需使用 VAD。	講話者離聽者不可太遠，且要在正前方。

表 5.1 演算法優缺點比較

Method	摘要
子空間演算	1.以 SVD 或 EVD 分解來區分語音子空間與雜訊子空間 2.語音訊息大多集中在較大的特徵值上 3.由壓低較小的特徵值來達到消雜訊
頻譜相減演算	1.以 FFT 將污染訊號轉至頻域 2.估計出雜訊在頻域上的分量 3.進而壓低這些分量 來減少雜訊對污染訊號的影響
卡爾曼演算	1.先出語音的 AR 係數 2.產生新的基底 3.新基底與 AR 係數線性組合來預測下一刻的乾淨語音
雙耳演算	1.計算雜訊到達左右耳的時間差 2.估計出雜訊的 PSD 3.左右耳需進行資料交換

表 5.2 演算法摘要

以下為適用於濾波器組架構下的演算比較，DCT 為助聽器計畫中另一個可以在濾波器組架構下使用的演算。

method	優點	缺點
Subtraction+18band	硬體結構簡單 計算量最低	1.PESQ 與 SNRseg 較 54band 來的低
Subtraction+54band	PESQ 與 SNRseg 皆最高	1.結構較為複雜 2.時間延遲為 18band 的四倍
kalman+18band	分頻過後，語音 PSD 變化較小，適合使用低階 AR 效果會較分頻前好	1.與 Subtraction+18band 相比，計算量較大
DCT on+18band	改善幅度大，語音區間的雜訊大多被消除	1.音樂效應較多 (白雜訊尤其嚴重)
DCT off+18band	同上	1.也有音樂效應 2.PESQ 與 DCT on 差不多 SNRseg 較 DCT on 低

表 5.3 演算法應用的優缺點比較

method	特性
Subtraction+18band Subtraction+54band	1. PESQ 與 SNRseg 皆較高 2. VAD 的判別好壞與突兀感無關 3. 可以增加語音辨識度 4. 語音區間仍就可以聽見背景雜訊
kalman+18band	1. PESQ 與 SNRseg 表現一般 2. VAD 的判別好壞與突兀感無關，但會影響消雜訊的效果 3. 對於雜訊的估計，要求較高 4. 沒有顯著的音樂效應
DCT on+18band	1. PESQ 與 SNRseg 皆較低 2. 消除背景雜訊效果相當顯著 3. 有較嚴重的音樂效應 4. 客觀量測的結果較低，應為音樂效應所致

表 5.4 演算法應用的特性比較

以下為頻譜相減演算 18 頻帶與卡爾曼演算 18 頻帶的計算量比較

Operation	加法	乘法
Weight	0	$(1+1/2+1/4+1/8+1/16+1/32)*3$
Py Pn estimate	$(1+1/2+1/4+1/8+1/16+1/32)*3$	$(1+1/2+1/4+1/8+1/16+1/32)*3$
合成	18	0
Total	24*24K	12*24K

表 5.5 頻譜相減演算 18 頻帶計算量(每秒)

Operation	加法	乘法
Measure diif	2	3
Generate gain	1	2
Generate x estimate	3	2
Generate x state	2	4
合成	18	0
Total	$(8*18+1)*24k$	$10*18*24k$

表 5.6 卡爾曼演算 18 頻帶的計算量(每秒)

5.3 展望

對頻譜相減演算與卡爾曼演算的研究工作，算是大致完成，而雙耳演算的則是處於比較初步的認識階段，但截止目前可以確定的是，它在估計雜訊方面有比較大的優勢，雙耳演算估雜訊不會受 VAD 的限制，可以獲得較完整的雜訊資訊。

但雙耳演算也有尚需克服的難處與代價，第一，在實作上，雙耳所接收到的資料要進行交換，如何傳輸與所需傳輸率為何，都需克服，第二，計算量方面，則多了計算雜訊到達左右耳的時間差，與一個類似零消的運算，

如果雙耳演算可以順利發展，我想它的實用價值應該不亞於 VAD 的使用，原先也沒有想過 VAD 的重要性，直到幾乎每個演算法都免不了要去區分語音與非語音區間時，才了解 VAD 的好壞往往也會主宰演算的效果，而 VAD 的目的也無非是想劃分非語音區間來估計雜訊，所以若能以雙耳演算來取代 VAD，我相信這樣可使大部分的演算法能有更好的效果。

參考文獻

- [1] Yi Hu and Philipos C. Loizou "A **generalized subspace approach for enhancing speech corrupted by colored noise**," *IEEE transactions on speech and audio processing* ,VOL.11 NO.4,2003
- [2] Sven Erik Nordholm and Ingvar Claesson "Spectral subtraction using reduced delay **convolution and adaptive averaging**, " *IEEE transactions on speech and audio processing* ,VOL.9 NO.8,2001
- [3] Wen Rong Wu and Po Cheng Chen "Subband **Kalman filtering for speech enhancement**," *IEEE transactions on circuit and system-II: analog and digital signal processing* ,VOL.45 NO.8,1998
- [4] Firas Jabloun, A. Enis Cetin and Engin Erzin "Teager energy based feature parameters for **speech recognition in car noise**" *IEEE signal processing letter* ,VOL.6 NO.10,1999
- [5] Jia Lin Shen ,Jeih Weih Hung and Lin Shan Lee "Robust entropy based endpoint detection for **speech recognition in noisy environments**," *Institute of information science* ,Academia Sinica Taipei ,Taiwan ,Republic of China
- [6] Chuan Jia and Bo Xu "An improved entropy based endpoint detection algorithm," National laboratory of pattern recognition, Institute of automation, Chinese academy of sciences, Beijing
- [7] Yu Ting Kuo, Tay Jyi Lin and Chih Wei Liu "Ultra low power **ANSI S1.11 filter bank for digital hearing aids**," Department of electronics engineering, National chiao Tung University, Taiwan soc Technology center, Industrial technology research institute, Taiwan
- [8] Yi Hu and Philipos C. Loizou "Evaluation of objective measures for **speech enhancement**," Department of electrical engineering university of Texas at Dallas Richardson, TX, USA

作者簡歷

姓 名：鍾譯賢

出 生 地：台灣省台北縣

出生日期：1980. 11. 30

學 歷：1988. 09 ~ 1994. 06	嘉義大學附設實驗小學
1994. 09 ~ 1997. 06	嘉義縣協同中學
1997. 09 ~ 2000. 06	嘉義縣協同中學
2001. 09 ~ 2005. 06	國立交通大學 電子物理學系 學士
2007. 09 ~ 2010. 03	國立交通大學 電子研究所系統組

