國 立 交 通 大 學

電信工程研究所

碩 士 論 文

應用盲訊號分離法於語音訊號分離之研究

A Study on Applying Source Separation Algorithms to Audio Signal Separation

研 究 生：黃靖雯　　Student: Ching-Wen Huang

指導教授：冀泰石 博士　　Advisor: Dr. Tai-Shih Chi

中華民國一百零一年三月

應用盲訊號分離法於語音訊號分離之研究

A Study on Applying Source Separation Algorithms to Audio Signal Separation

研 究 生：黃靖雯　　Student: Ching-Wen Huang

指導教授：冀泰石 博士　　Advisor: Dr. Tai-Shih Chi

國立交通大學

電信工程研究所

碩士論文

A Thesis
Submitted to Institute of Communication Engineering
College of Electrical and Computer Engineering
National Chiao-Tung University
In Partial Fulfillment of the Requirements
for the Degree of
Master of Science in
Communication Engineering
March 2012
Hsin-Chu, Taiwan, Republic of China

中華民國一百零一年三月

應用盲訊號分離法於語音訊號分離之研究

學生：黃靖雯　　指導教授：冀泰石 博士

國立交通大學電信工程研究所

感知訊號處理實驗室

摘要

本論文探討盲訊號分離法對於語音分離的成效，主要考量經頭部相關傳輸函數（Head-Related Transfer Function, HRTF）為脈衝響應的摺積混和訊號。此舉是為模擬音源來自不同方位時，人的左右耳所聽到的聲音，再利用雙耳所收到的混和訊號之間的差異來做分離。論文中實驗了三種不同的方法，第一種方法為影像分離演算法，將混和訊號的聲譜圖作為影像訊號來處理，利用影像的邊緣圖具有稀疏的特性，以最大化影像邊緣圖作為演算法的目標，並以 FastICA 計算疊代法的初始條件。第二種方法加入稀疏成分分析法的考量，利用非線性投影將影像的邊緣訊號轉成較為稀疏的訊號，作為影像分離演算法的前置處理。第三種方法將聲譜圖分頻，利用每個頻帶的混和訊號稀疏的特性，使用非線性函數計算音源對於混和訊號的貢獻值，用非線性遮蔽的方法將音源抽取出來。實驗結果顯示，當兩個音源分別位於左右兩邊時，三種方法皆可達到良好的分離效果；當音源來自同一邊的混和情況，所提出的第三種方法也可成功分離音源，與其他現有演算法的比較結果也顯示此方法的方離效果良好且穩定。

A Study on Applying Source Separation Algorithms to Audio Signal Separation

Student: Ching-Wen Huang Advisor: Dr. Tai-Shih Chi

Institute of Communication Engineering

National Chiao-Tung University

Perception Signal Processing Laboratory

Abstract

In this thesis, we propose audio separation algorithms stemmed from blind source separation algorithms. The Head-Related Transfer Functions (HRTFs) are used to simulate the convolutive mixing of audio sources to model the sound mixtures perceived by listeners. Three different methods are proposed and investigated. The first method is an image separation method named ISBS, in which we consider audio spectrograms as images. The criterion of this algorithm is to maximize the sparsity of the edges of the signals and the FastICA algorithm is used to initiate its iteration. The second method adopts a nonlinear projection as the pre-processing of the ISBS algorithm. The nonlinear projection transforms the edges onto a sparser domain. The third method considers the frequency bin-wise mixtures and utilizes the sparsity in the time-frequency (T-F) domain. For each T-F unit, we calculate the contributions from each source by a nonlinear function. A masking matrix is formed based on contributions from each source and used to extract original sounds. Simulation results showed that all three methods performed well when sound sources located far apart from each other. When sources are at close locations, only the proposed third method performed well. Comparison with some conventional methods also showed the third method performed better and more robust in most cases.

目　　錄

表 目 錄

圖 目 錄

第一章　緒論

1.1 研究背景

在日常生活中，常常會遇到多個聲源同時出現的情況，例如在一個吵嘈的環境中，語音會夾雜著各式的噪音如手機鈴聲、汽機車噪音等，或者是在同一個空間中，有很多人同時說話，但我們通常都只想要集中在某個人的聲音上面，要如何從吵雜的混和訊號中抽取出我們想要的訊號源，就是所謂的雞尾酒派對問題（Cocktail-Party Problem）。

盲訊號分離法（Blind Source Separation，BSS）為近十幾年來很熱門的一個研究主題，所謂「盲」指的就是我們只有收到的混和訊號（mixtures），訊號源（sources）和混和的過程（mixing process）皆為未知，目標就是在只有混和訊號的情況下，分離出原本的訊號源。因此盲訊號分離法廣泛地應用於未知訊號的處理，如在生醫訊號處理（Biomedical Signal Processing）的應用，在量測到的腦電波訊號（Electroencephalogram , EEG）中，可能混和著肌肉運動、眼球活動、心臟跳動等訊號源，不同位置所量測到的 EEG 訊號即為混和訊號，BSS 的目的便是從這些混和訊號中分離出原本的訊號源。其他應用還有特徵擷取（Feature Extraction）、通訊（Telecommunication）、金融序列分析（Financial Time Series Analysis）等[1][2]，在語音訊號處理（Audio Signal Processing）上的應用大部分為語音分離（Audio Separation），也就是用來解決前段所提的雞尾酒派對問題，在許多聲源的混和音訊中，個別分離出每個聲音。

1.2 盲訊號分離法簡介

盲訊號分離法的基本架構如（圖 1-1）所示，假設有兩個聲音源 s_1、s_2，經過了一個未知的混和過程，兩個麥克風所收到的訊號就是兩個聲源的混和訊號 x_1、x_2，可以寫成下式：

$$x_1 = a_{11}s_1 + a_{12}s_2$$
$$x_2 = a_{21}s_1 + a_{22}s_2 \quad (1\text{-}1)$$

其中 a_{11}、a_{12}、a_{21}、a_{22} 取決於聲源到麥克風的距離，代表聲源到麥克風的衰減狀況，這些係數皆為未知，盲訊號分離法的目標就是在混和過程與訊號皆未知的情況下，從混和訊號中分離出訊號源。

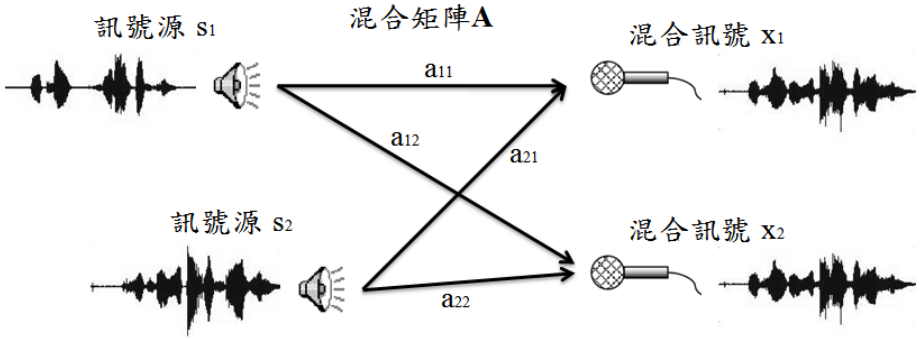


圖 1-1　盲訊號分離法示意圖

1.2.1 即時性混和（instantaneous mixing）

將盲訊號分離法的數學表示法如下，假設有 N 個訊號源 $s_1 \ldots s_N$，M 個接收器收到 M 個混和訊號 $x_1 \ldots x_M$，寫成向量模式即為一個 N x 1 的訊號源向量 $\mathbf{s}=[s_1 \ldots s_N]^T$，M x 1 的混和訊號向量 $\mathbf{x}=[x_1 \ldots x_M]^T$，最基本的混和過程假設就是即時性混和（instantaneous mixing），也就是如（圖 1-1）所示，混和訊號 **x** 為訊號源 **s** 和一個混和矩陣（mixing matrix）**A** 相乘，其中 **A** 為一個 M x N 的矩陣，如下式：

$$\mathbf{x}=\mathbf{A}\mathbf{s} \quad (1\text{-}2)$$

盲訊號分離法的目的就是找出一個分離矩陣（demixing matrix）**W**，使此分離矩陣和混和訊號 **x** 相乘後，可以分離出訊號源 **s**（式 1-3）。可看出若是解出的分離矩陣 **W** 越接近 **A** 的反矩陣 $\mathbf{A}^{-1}$，**y** 就會越接近原本的訊號源 **s**。

$$\mathbf{y} = \mathbf{W}\mathbf{x} = \hat{\mathbf{s}} \quad (1\text{-}3)$$

要在只有混和訊號 **x** 的情況下，分離出矩陣 **A** 和訊號源 **s**，也就是對 **x** 的矩陣分解，在 **A** 與 **s** 都沒有特別限制的情況下，是沒有辦法得到唯一解的，因此通常都會假設訊號源滿足某種統計特性，利用這種統計特性來做分離，根據不同的假設，就發展出了不同的演算法。如獨立成分分析法（Independent Component Analysis, ICA）便是假設訊號源彼此是獨立的（independent）[1-9]，從混和訊號中抽取出獨立的成分；稀疏成分分析法（Sparse Component Analysis, SCA）假設訊號源是非常稀疏的（sparse）[17-20]；非負矩陣分解法（Non-negative Matrix Factorization, NMF）則是用在訊號源皆為非負值的假設之下[21][22]，詳細的方法會在之後的章節裡介紹。

1.2.2 摺積混和（convolutive mixing）

最初 BSS 的基本假設很簡單，混和訊號只是訊號源與混和矩陣 **A** 的相乘，也就是訊號源與矩陣係數的相加相乘（式 1-1），但在實際錄音環境下，聲音源到麥克風之間除了直接路徑（direct path）以外，還會經由不同的反射路徑而產生不同的延遲（delay）與衰減（圖 1-2），所以麥克風所收到的混和訊號其實是經過摺積混和（convolutive mixing）的訊號，如（式 1-3）所示，其中 a_{ik} 代表第 k 個訊號源到第 i 個麥克風的脈衝響應（impulse response），τ 為時間延遲。

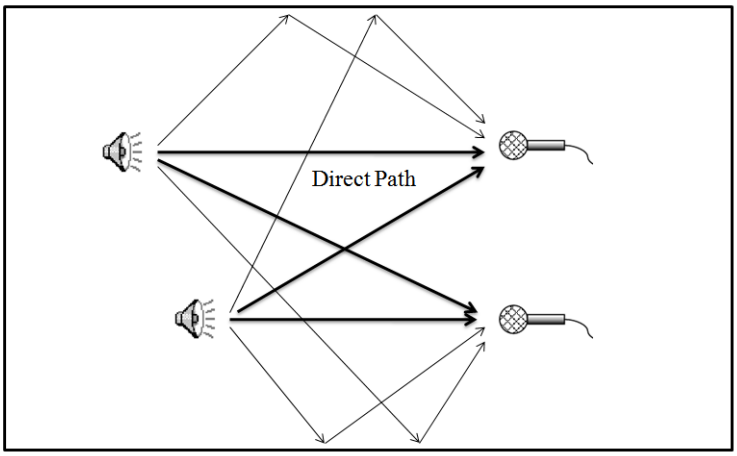


圖 1-2　摺積混和訊號示意圖

$$x_i(t) = \mathbf{A} * \mathbf{s}(t) = \sum_k \sum_{\tau=0}^{\infty} a_{ik}(\tau) s_k(t-\tau) \quad (1\text{-}3)$$

$$\mathbf{X}(\omega,\tau) = \mathbf{A}(\omega)\mathbf{S}(\omega,\tau) \quad (1\text{-}4)$$

$$\mathbf{X}(\omega,\tau) = \sum_\tau x(t) w(t-\tau) e^{-j\omega\tau} \quad (1\text{-}5)$$

因此，原本只須解出一個分離矩陣 **W**，在此情況下卻變成了一個反摺積（deconvolution）的問題。針對摺積混和的情況，P. Smaragdis 於 1998 年、S. Ikeda 於 1999 年提出了 Convolutive ICA[13][14]，主要概念就是將原本的時域（time domain）訊號轉到頻域（frequency domain），如此一來在時域上的摺積就會在頻域上變成相乘（式 1-4），因此將混和訊號經過 Short-term Fourier Transform（STFT）後（式 1-5），對於每個頻帶的成分而言都是即時性混和，此時就可以對每個頻帶的訊號分別使用即時性混和的分離演算法，最後再用 Inverse STFT 重建回聲音，流程圖如（圖 1-3）[17]。

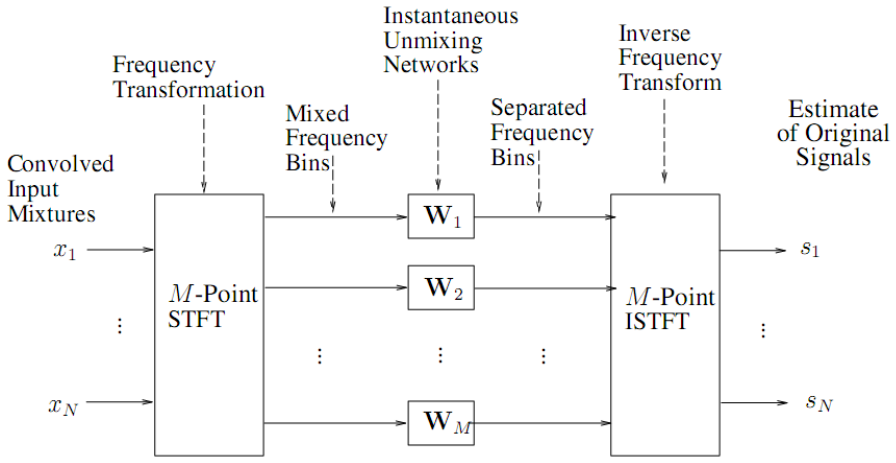


圖 1-3　Convolutive ICA 流程圖

將摺積混和的訊號轉到頻域，再對每個頻率分別做 ICA，為目前很常見的一個方法[13-16]，但此作法一直存在著一些潛在問題如排列問題（permutation）和縮脹問題（dilation）。如（圖 1-4）所示，對每個頻帶的訊號都分別視為一個即時性混和的問題來解，每次解的過程也是互相獨立的，因此每個頻帶所解出來的訊號順序及大小會不一樣，如此在重建回聲音時，就會產生排列問題和縮脹問題。另外，訊號做了傅立葉轉換（Fourier Transform）後會有虛數部分，因此若要直接於每個頻帶應用即時性混和的演算法，也必須將演算法發展成適合虛數計算，如 Complex FastICA[7]。

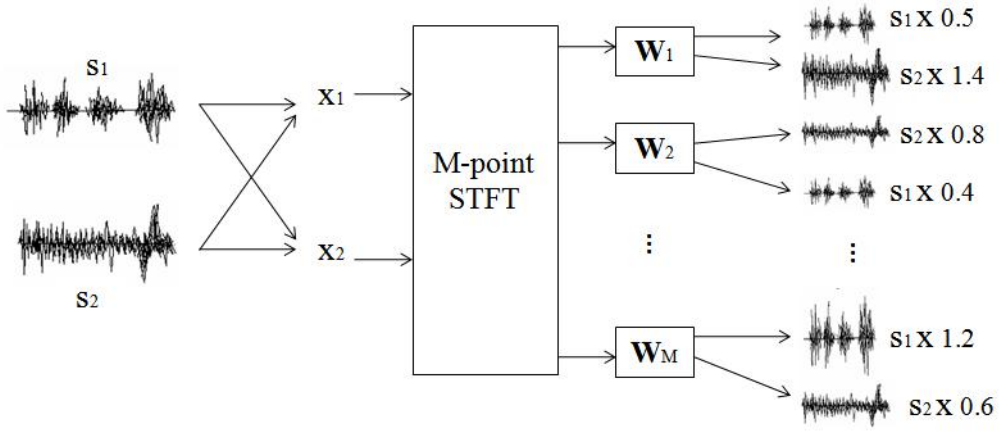


圖 1-4　排列問題及膨脹問題

1.2.3 欠定問題（unter-determined problem）

盲訊號分離法中還有另一個問題，也就是訊號源的數量 n 與混和訊號的數量 m 可能會不相等。一般最簡單的假設是 M=N，稱為 even-determined problem，此時 **A** 為方陣，$\mathbf{A}^{-1}$ 存在，可以直接將 $\mathbf{A}^{-1}$ 與混和訊號 **x** 相乘；當 M>N 時稱為 over-determined problem，雖然無法直接求得 $\mathbf{A}^{-1}$，但可用 pseudo-inversion。而當訊號源數量大於混和訊號數量，也就是 M<N 時，稱為 unter-determined problem，此時就無法直接用矩陣計算將訊號源解回來，因此 ICA 演算法通常都會假設混和訊號的數量要大於或等於訊號源的數量(M ≥ N)。

1.3 研究方法

本論文主要針對在實際環境下，人的左右耳所收到的混和訊號來做分離，我們使用頭部相關傳輸函數（Head-Related Transfer Function, HRTF）作為摺積混和的脈衝響應，也就是模擬當聲音源來自某個方位角時，雙耳所聽到的聲音，利用雙耳所收到的混和訊號之間的差異來做分離。我們之所以考慮 HRTF 混和訊號，是希望可以將盲訊號分離法應用於雙耳助聽器，例如將混和訊號中的語音訊號與噪音個別分離出來，便可以保留語

音訊號而捨棄噪音，達到雙耳麥克風降噪的效果。在本研究中，我們首先對於即時性混和的情況下成功地嘗試了現有的演算法，之後對於此種較為複雜的摺積混和情況，提出以聲譜圖作為影像訊號所改善的演算法。

1.4 章節大綱

本論文的各章內容如下：第二章為獨立成分分析法的背景簡介，並介紹兩種主要的 ICA 演算法，Infomax 與 FastICA；第三章介紹論文所使用的影像分離演算法－Iterative Sparse Blind Separation，ISBS，以及其在聲譜圖上的實際應用；第四章將介紹稀疏成分分析法，以及我們利用非線性投影量所提出的遮蔽方法；第五章為實驗結果與相關方法的比較，以及相關討論與結論。

第二章　獨立成分分析法

2.1 發展背景

獨立成分分析法的想法最早是由 J. Herault 與 C. Jutten 於 1986 年所提出，當初是以一個簡單的適應性演算法用在類神經網路架構（neural network）上來分離訊號，但並沒有提出正式的數學理論，之後是在 1994 年由 P. Comon 提出公式定義與證明，並推導出以最小化混和訊號的共同資訊（mutual information）作為目標函數，這是 ICA 最早期的歷史[3]。與此同時，1992 年由 Linsker 發展的 Infomax，是一種基於消息理論（Information Theory）的非監督式學習法（Unsupervised learning rule），主要目的是要最大化類神經網路的輸入與輸出之交互資訊；而在 1994 年，Nadal 與 Parga 發現當雜訊很小時，在輸入與輸出的共同資訊為最大時，輸出的訊號會彼此獨立。因此在 1995 年 **J.** Bell 與 J. Sejnowski 便將這種 Infomax 的技術應用於 ICA 上面，提出了一種有效率的演算法，稱為 InfomaxICA[8][9]。其他的近似方法也有如 Gaeta 與 Lacoume 於 1990 年提出以最大概似估計法（Maximum Likelihood Estimation , MLE）作為演算法計算[10]，而在 1996 年 Pearlmutter 與 Parra 證明出 MLE 與 Infomax 其實是一樣的[11]。但最原始的 InfomaxICA 只能用來分離超高斯（super-Gaussian）訊號，因此在 1999 年，T.W. Lee 與 Girolami 將 InfomaxICA 繼續做延伸，發展了可用於 sub-Gaussian 與 super-Gaussian 的演算法[12]。

除了以共同資訊作為獨立性的量測外，另一類很熱門的方法為 A. Hyvärinen 與 E. Oja 於 1999 年所提出的 FastICA[4][5]，是以訊號的常態性（normality）作為獨立性的判斷標準，利用語音訊號為超高斯分布的特性，以最大化混和訊號的峰值（kurtosis）作為演算法目標，並用 fixed-point 演算法求最佳解，收斂速度很快是其優點，缺點是峰值的計算容易受偏離值（outlier）所影響，作為常態性的量測有時並不穩定。本論文使用 FastICA 作為影像分離演算法的初始條件計算，因此收斂速度快、不影響演算法速度為主要考量，以下將簡介 InfomaxICA 的基本運算及介紹 FastICA 的詳細運算過程。

2.2 InfomaxICA

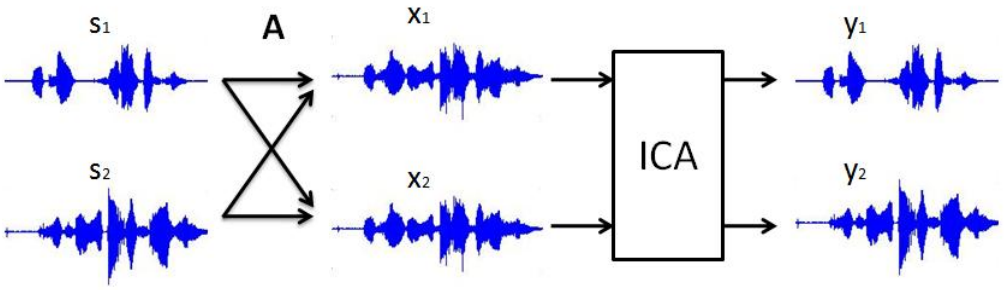


圖 2-1　ICA 基本架構

ICA 的基本架構如（圖 2-1），訊號源 s_1 與 s_2 經過一個未知的混和矩陣 **A**，得到兩個混和訊號 x_1 與 x_2，在經過 ICA 的運算後，希望分離出的訊號 y_1 與 y_2 就是原本的訊號源 s_1 與 s_2。

InfomaxICA 主要是用共同資訊做為訊號獨立性的量測，目標是希望能最小化 y_1 與 y_2 的共同資訊。共同資訊的計算如（式 2-1），$I(y_1,y_2)$代表 y_1 與 y_2 的共同資訊，$H(y_1)$為訊號 y_1 的熵(entropy)，為 y_1 的不確定性，熵的定義如（式 2-2），其中 p_i 代表 y 的機率。

$$I(y_1, y_2) = H(y_1) + H(y_2) - H(y_1, y_2) \tag{2-1}$$

$$H(y) = -\sum_i p_i(y)\log_2 p_i(y) \tag{2-2}$$

$$H(y_1, y_2) = H(y_1) + H(y_2|y_1)$$

$$= -\sum_i p_i(y_1)\log_2 p_i(y_1) - \sum_i p_i(y_2|y_1)\log_2 p_i(y_2|y_1) \tag{2-3}$$

由（式 2-1）可看出要最小化 $I(y_1,y_2)$，表示要最大化 $H(y_1,y_2)$，也就是 y_1 與 y_2 的聯合熵（joint entropy）（式 2-3），而當 y_1 與 y_2 為獨立時，$p(y_1,y_2)= p(y_1)p(y_2)$，$p(y_2|y_1)=p(y_2)$，$H(y_2|y_1)$就會等於 $H(y_2)$。而根據消息理論，$H(y_2|y_1) \leq H(y_2)$，因為在 y_1 事件發生時，y_2 也發生的不確定性會比 y_2 整體的不確定要來的小，所以當 y_1 與 y_2 彼此獨立時，$H(y_1,y_2)$有最大值，$I(y_1,y_2)$有最小值。從另一個角度來說，當 y_1 與 y_2 為獨立時，表示 y_1 與 y_2 沒有共同的資訊，$I(y_1,y_2)$為 0，因此最小化 $I(y_1,y_2)$可以作為獨立訊號分離的目標函數。

2.3 FastICA

FastICA 則是以訊號的分佈狀態作為獨立性的量測，通常語音訊號都會有多數的沉默片段，所以在統計直方圖（histogram）上大部分的值都會在 0 附近，屬於超高斯分布（Super-Gaussian Distribution）(圖 2-2)，經過混和之後，根據中央極限定裡(Central Limit Theorem)，當資料數越來越多時，混和的訊號會越趨近於高斯分布（Gaussian Distribution)。(圖 3-3)顯示 16 個語音訊號的統計直方圖，可以看出混和成 2 個訊號後，混和訊號會傾向高斯分布（圖 2-3)。因此 FastICA 的目的便是從混和訊號中分離出 Non-Gaussian 的訊號，也就是用常態性（normality）作為獨立性量測的依據。

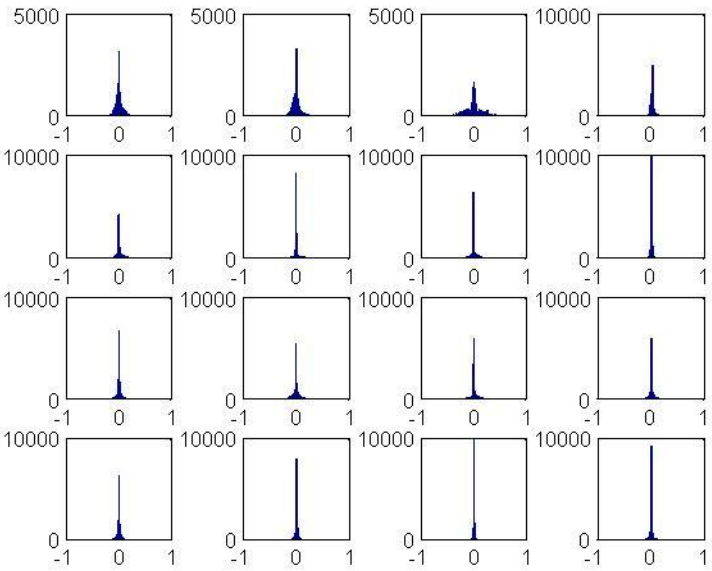


圖 2-2　16 個訊號源的統計直方圖

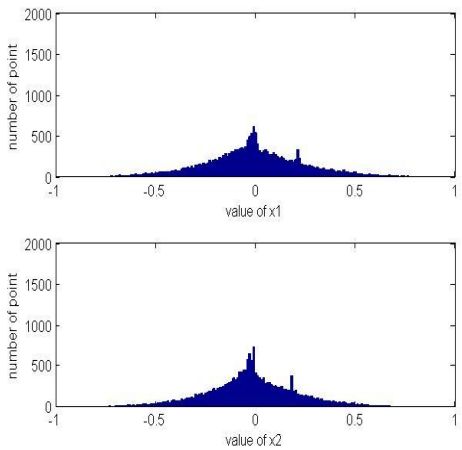


圖 2-3　兩個混和訊號的統計直方圖

FastICA 是利用峰值（Kurtosis）來計算 normality，Kurtosis 為四階動差，對不同的高斯分布（圖 2-4）會有不同的值（式 2-4）

$$\mathrm{Kurt}(y) = \frac{\frac{1}{N}\sum_{t=1}^{N}(\bar{y}-y^t)^4}{\left(\frac{1}{N}\sum_{t=1}^{N}(\bar{y}-y^t)^2\right)^2} - 3 = \frac{E[(\bar{y}-y^t)^4]}{(E[(\bar{y}-y^t)^2])^2} - 3 \begin{cases} = 0 & \text{for Gaussian} \\ > 0 & \text{for super} - \text{Gaussian} \\ < 0 & \text{for sub} - \text{Gaussian} \end{cases} \tag{2-4}$$

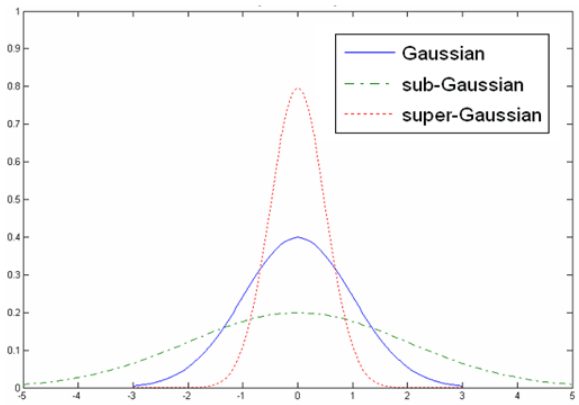


圖 2-4　不同的高斯分布

因此 Kurtosis 值越大，表示訊號越趨近超高斯分佈，也就越趨近於獨立，FastICA 便以最大化 Kurtosis 為演算法目標。

為了簡化演算法計算，降低需要計算的變數量，我們通常都會假設獨立的訊號源都是 zero-mean、unit-variance，但實際應用上，收到的混和訊號卻不一定保有這兩種特性，因此會先對訊號對一些前置處理，如集中變數（centering）以及資料白化（whitening），使訊號先變成非相關性（un-correlated），之後再解分離矩陣。

2.3.1 前置處理

集中變數(centering)的作法只要將收到的混和訊號**x**減掉它的期望值E[**x**](式2-5)，就可以使訊號的平均值為0。

$$\mathbf{x}=\mathbf{x}-E[\mathbf{x}] \tag{2-5}$$

白化（whitening）的目的是使訊號變成 unit-variance，也就是 $E[\mathbf{x}\mathbf{x}^T]=1$，代表 X 的成分是非相關的，其作法如下：

首先對 **x** 的共變異數矩陣（covariance matrix）作特徵值分解（eigenvalue decomposition）（式 2-6），分解成由特徵值（eigenvalue）所組成的對角矩陣 **D**，以及特徵向量（eigenvector）所組成的矩陣 **V**，同時定義白化矩陣 **U** 如（式 2-7）。

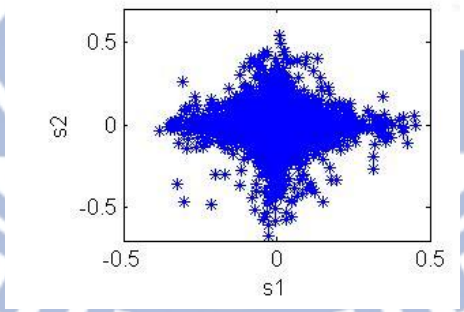
$$R_X = E[\mathbf{x}\mathbf{x}^T] = \mathbf{D}\mathbf{V} \tag{2-6}$$

$$\mathbf{U} \triangleq \mathbf{D}^{-\frac{1}{2}}\mathbf{V}^T \tag{2-7}$$

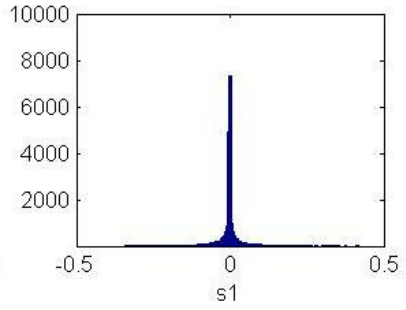
如此一來白化訊號（whitening signal）**z** 就是白化矩陣 **U** 與原訊號 **x** 相乘（式 2-8），而 **z** 的共變異數矩陣就會變成單位矩陣（式 2-9）。

$$\mathbf{z} \triangleq \mathbf{U}\mathbf{x} = \mathbf{D}^{-\frac{1}{2}}\mathbf{V}^T\mathbf{x} \tag{2-8}$$

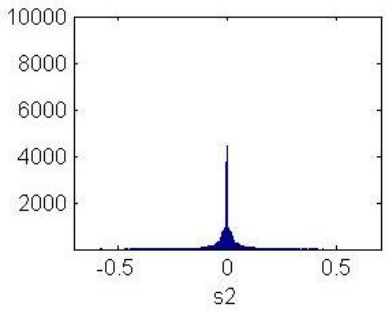
$$E[\mathbf{z}\mathbf{z}^T] = \mathbf{D}^{-\frac{1}{2}}\mathbf{V}^T E[\mathbf{x}\mathbf{x}^T]\mathbf{V}\mathbf{D}^{-\frac{1}{2}} = \mathbf{D}^{-\frac{1}{2}}\mathbf{V}^T\mathbf{V}\mathbf{D}\mathbf{V}^T\mathbf{V}\mathbf{D}^{-\frac{1}{2}} = \mathbf{D}^{-\frac{1}{2}}\mathbf{D}\mathbf{D}^{-\frac{1}{2}} = \mathbf{I} \tag{2-9}$$



(a) 訊號源 s_1 與 s_2 的散佈圖



(b) s_1 的統計直方圖



(c) s_2 的統計直方圖

圖 2-5　訊號源的散佈圖及統計直方圖

訊號源經過混和與 FastICA 處理後的變化如（圖 2-5）~（圖 2-6）所示，（圖 2-5(a)）為訊號源的散佈圖（scatter plot），橫軸為音源訊號 s_1 的值，縱軸為音源訊號 s_2 的值，當中的點座標為每個時間點 s_1 與 s_2 的值，也就是$(s_1(t), s_2(t))$，因為訊號源為超高斯分佈，聚集在 0 附近的點很多，所以兩個訊號的分佈圖會呈現一個十字的狀態；（圖 2-5(b)）為(a)圖對橫軸投影後的統計分佈圖，也就是音源 1 的統計直方圖，（圖 2-5(c)）為(a)圖對縱軸投影後的統計分佈圖，也就是音源 2 的統計直方圖。

訊號源經過混和後的訊號散佈圖如（圖 2-6(a)），(b)圖為經過集中變數後的結果，在此因為原本的混和訊號就是 zero-mean，所以與(a)圖無異。(c)圖為經過白化的訊號，可看出 z_1 與 z_2 為非相關的，而分離矩陣則是進一步將非相關的訊號轉成獨立的訊號(d)，由(c)到(d)圖可看出分離矩陣其實就是一個旋轉矩陣。

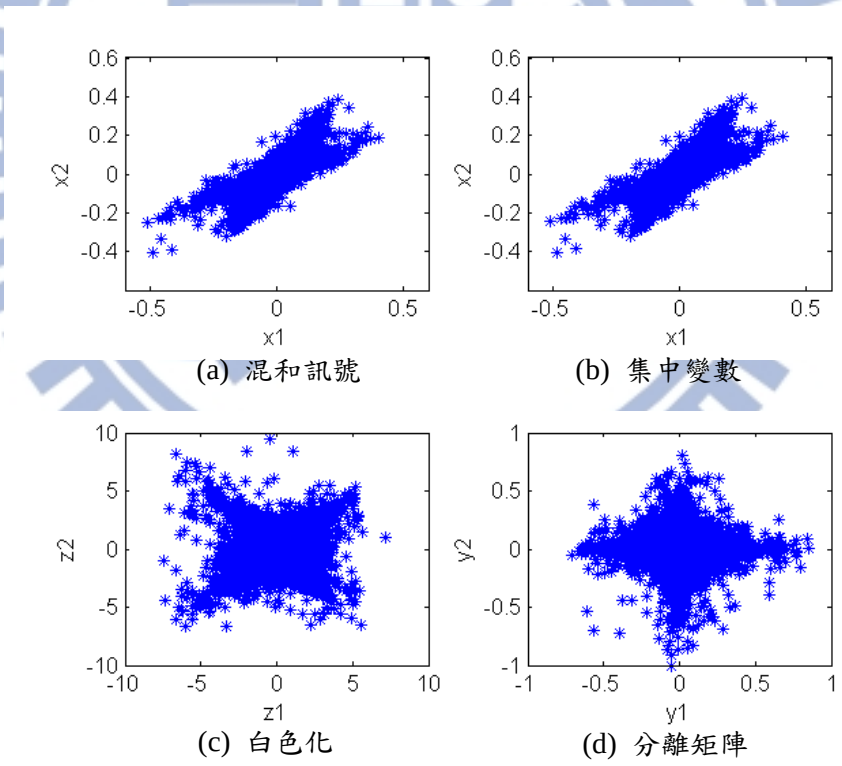


圖 2-6　訊號經過 FastICA 處理後的變化

2.3.2 解分離矩陣

前面提到 FastICA 是以最大化 Kurtosis 作為演算法目標，對 zero-mean、unit-variance 的訊號而言，Kurtosis 的公式可簡化為（式 2-10）。

假設分離矩陣為 $\mathbf{W}$，當中的每個行向量 $\mathbf{w_i}$ 可以解出一個訊號源，我們希望可以最大化 $\mathbf{w}_i^T\mathbf{Z}$ 的 kurtosis（式 2-11）。因為 $\mathbf{W}$ 實為一個旋轉矩陣，我們假設 $\mathbf{w_i}$ 的 norm 皆為 1。

$$\text{kurt}(y) = E[y^4] - 3(E[y^2])^2 \tag{2-10}$$

$$\text{kurt}(\mathbf{w}_i^T\mathbf{z}) = E\left[(\mathbf{w}_i^T\mathbf{z})^4\right] - 3\left(E\left[(\mathbf{w}_i^T\mathbf{z})^2\right]\right)^2 = E\left[(\mathbf{w}_i^T\mathbf{z})^4\right] - 3\|\mathbf{w}_i\|^4 \tag{2-11}$$

$$\nabla_w \text{kurt}(\mathbf{w}_i^T\mathbf{z}) = E\left[\mathbf{z}(\mathbf{w}_i^T\mathbf{z})^3\right] - 3\|\mathbf{w}_i\|^2\mathbf{w}_i \tag{2-12}$$

對（式 2-11）取 gradient 會得到（式 2-12），因為$\|\mathbf{w}\|=\|\mathbf{z}\|=1$，$\nabla_w \text{kurt}(\mathbf{w}_i^T\mathbf{z}) < 1$，根據 fixed-point 演算法，我們可以得到解 $\mathbf{w}_i$ 的疊代公式如（式 2-13）。

$$\mathbf{w}_i(k) = E[\mathbf{z}(\mathbf{w}_i(k-1)^T\mathbf{z})^3] - 3\mathbf{w}_i(k-1) \tag{2-13}$$

FastICA 的整體計算流程如下：

(1) 集中變數，使訊號變成 zero-mean。

(2) 白化處理，使訊號變成 un-correlated。

(3) 隨機選擇分離矩陣 $\mathbf{W}$ 的初始條件，設定 i=1，k=1。

(4) 計算$\mathbf{w}_i(k) = E[\mathbf{z}(\mathbf{w}_i(k-1)^T\mathbf{z})^3] - 3\mathbf{w}_i(k-1)$

(5) 使 $\mathbf{w_i}$ 變成 unit-norm，$\mathbf{w}_i(k) = \frac{\mathbf{w}_i(k)}{\|\mathbf{w}_i(k)\|}$

(6) 若$\left|\left|\mathbf{w}_i(k)^T\mathbf{w}_i(k-1)\right| - 1\right| \gg 0$，表示 $\mathbf{w_i}$ 尚未收斂，繼續設定 k=k+1，回到(4)再次疊代。

第三章　影像分離演算法

盲訊號分離法除了用在語音訊號上之外，也有在影像訊號上的應用，如天文影像圖分離、文字文件重建、fMRI 腦血流量圖分析等，大部分的作法是將二維的影像訊號重新排列成一維訊號，再使用現有的一維訊號演算法，不過此種作法會破壞影像訊號相鄰圖素（pixel）間的相關性，因此也有一些演算法直接針對二維訊號做處理[23]。另外也有利用影像訊號皆為非負值的特性，使用非負矩陣分解法（Non-negative Matrix Factorization, NMF）將影像分離的問題轉換成矩陣分解問題，在演算法內加上訊號源統計特性的限制，也可以達到影像分離的效果[21][22]。

在 ICA 對語音分離的實驗中，我們確實驗證了演算法在即時性混和的情況下有很好的分離結果，但對於摺積混和訊號就無法分離。考慮將訊號轉到頻域的作法較為複雜，並且會產生排列問題及縮脹問題等，我們嘗試將語音訊號的聲譜圖（spectrogram）當作一個影像訊號，使用影像分離演算法將混和訊號的聲譜圖分離出來，再重建回聲音。

我們所使用的演算法為 Iterative Sparse Blind Separation（ISBS）[24]，主要是利用影像的邊緣（edge）會比較稀疏的特性，使用ℓ-2 norm 作為稀疏性（sparsity）的量測，以最大化混和訊號之邊緣的ℓ-2 norm 作為演算法的目標。實驗中發現疊代法（iteration）的初始條件會對分離效果有很大的影響，因此我們加入了 FastICA 作為初始條件的計算，以下將詳細介紹。

3.1 Notation and Model

假設有 N 張大小為m_s x n_s的影像圖 S_1….S_N，經過混和矩陣 A 得到 M 張混和影像 X_1…X_M，寫成矩陣形式如下：

$$\mathbf{X}(m,n) = \mathbf{A}\mathbf{S}(m,n) \tag{3-1}$$

則 $\mathbf{S}(m,n)=[S_1(m,n)\ldots S_N(m,n)]$為一個 Nx1 的向量，$\mathbf{X}(m,n) =[X_1(m,n)\ldots X_M(m,n)]$為 Mx1 的向量，**A** 為 M x N 的混和矩陣。目標就是找出一個分離矩陣 **B**，使其乘上混和的影像 $\mathbf{X}(m,n)$後可以分離出原本的影像 $\mathbf{S}(m,n)$（式 3-2）。

$$\hat{\mathbf{S}}(m,n) = \mathbf{B}\mathbf{X}(m,n) \tag{3-2}$$

3.2 Sparse Representation

如同 FastICA 將語音訊號視為超高斯分佈，以最大化峰值（kurtosis）作為最佳化演算法的目標，ISBS 也是先將影像訊號轉到一個稀疏的領域，再以最大化稀疏性作為最佳化目標。

我們使用拉普拉斯轉換（Laplacian Trasform）作為稀疏域的轉換：

$$\boldsymbol{\mathcal{S}} = \nabla\mathbf{S} = \frac{\partial^2\mathbf{S}}{\partial^2 x} + \frac{\partial^2\mathbf{S}}{\partial^2 y} \tag{3-3}$$

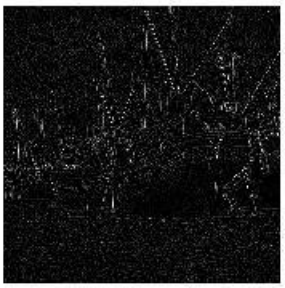
寫成離散的形式：

$$\boldsymbol{\mathcal{S}}(m,n) = \mathbf{S}(m+1,n) + \mathbf{S}(m-1,n) + \mathbf{S}(m,n+1) + \mathbf{S}(m,n-1) - 4\mathbf{S}(m,n) \tag{3-4}$$

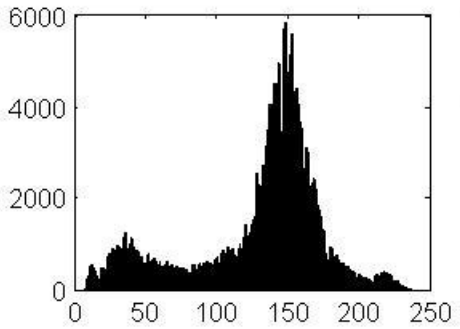
可以看出對影像訊號做拉普拉斯轉換，相當於取出一張影像的邊緣部分。選擇拉普拉斯轉換一來是邊緣訊號會比原本的影像訊號更為稀疏，如（圖 3-1）所示，且聲譜圖的邊緣訊號也包含了最重要的諧音（harmonic）資訊；二來是因為拉普拉斯轉換為線性轉換，如此一來在邊緣圖上解出的分離矩陣就可以用回原本的影像訊號。混和訊號經過拉普拉斯轉換後，表示如下：

$$\boldsymbol{\mathcal{X}} = \frac{\partial^2\mathbf{A}\mathbf{S}}{\partial^2 x} + \frac{\partial^2\mathbf{A}\mathbf{S}}{\partial^2 y} = \mathbf{A}\boldsymbol{\mathcal{S}} \tag{3-5}$$

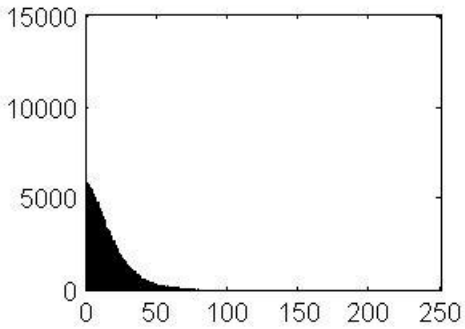
(a)原影像



(b)邊界圖



(c)原影像之統計直方圖



(d)邊界圖之統計直方圖

圖 3-1　影像與邊界圖及其統計直方圖

3.3 Algorithm

我們使用ℓ-2 norm 作為稀疏性的量測，定義如下：

$$G(\boldsymbol{S}) = \sum_{i=1}^{N}[\mathcal{J}(\mathcal{S}_i)]^{\frac{1}{2}} \tag{3-6}$$

$$\mathcal{J}(\mathcal{S}_i) = \frac{1}{m_s n_s}\sum_{m=1}^{m_s}\sum_{n=1}^{n_s}|\mathcal{S}_i(m,n)|^2 \tag{3-7}$$

將混和的影像訊號 **X**(m,n)利用拉普拉斯轉換成邊緣圖 **E**(m,n)，再將 **E**(m,n)經過白化處理使其變成非相關性訊號 **Z**(m,n)。定義$\boldsymbol{\mathcal{H}}(m,n) \triangleq \mathbf{BZ}(m,n) = \widehat{\boldsymbol{S}}(m,n)$為分離影像的拉普拉斯轉換，演算法的目標為找到一個分離矩陣 **B**，使$\boldsymbol{\mathcal{H}}(m,n)$回復為原影像的邊緣圖。B 滿足下式：

$$\mathbf{B} = \min_{\mathbf{B}}\{G(\boldsymbol{\mathcal{H}})\} \tag{3-8}$$

我們使用（式 3-9）來疊代計算分離矩陣 **B**，同時更新$\boldsymbol{\mathcal{H}}(m,n)$的值如式(3-10)。

$$\mathbf{B}^{(k+1)} = (\mathbf{I} + \boldsymbol{\epsilon}^{(k)})\mathbf{B}^{(k)} \tag{3-9}$$

$$\boldsymbol{\mathcal{H}}^{(k+1)}(m,n) = (\mathbf{I} + \boldsymbol{\epsilon}^{(k)})\boldsymbol{\mathcal{H}}^{(k)}(m,n) \tag{3-10}$$

將（式 3-7）代入（式 3-10）可得到（式 3-11）。假設每次的更新值$\epsilon_{ij}^{(k)}$都是很小的，也就是$\left|\epsilon_{ij}^{(k)}\right| \ll 1$，如此一來（式 3-11）可展開成（式 3-12）

$$\mathcal{J}\left(\mathcal{H}_i^{(k+1)}\right) = \frac{1}{m_s n_s}\sum_{m=1}^{m_s}\sum_{n=1}^{n_s}\left|\mathcal{H}_i^{(k)}(m,n) + \sum_{j=1}^{N}\epsilon_{ij}^{(k)}\mathcal{H}_j^{(k)}(m,n)\right|^2 \tag{3-11}$$

$$\mathcal{J}\left(\mathcal{H}_i^{(k+1)}\right) \approx \frac{1}{m_s n_s}\sum_{m=1}^{m_s}\sum_{n=1}^{n_s}\left\{\left|\mathcal{H}_i^{(k)}(m,n)\right|^2 + 2\sum_{j=1}^{N}\epsilon_{ij}^{(k)}\left(\left|\mathcal{H}_i^{(k)}(m,n)\right|\mathrm{sgn}\left(\mathcal{H}_i^{(k)}(m,n)\right)\mathcal{H}_j^{(k)}(m,n)\right)\right\} \tag{3-12}$$

為簡化表示，定義矩陣 **R** 與矩陣 **D** 如（式 3-13、3-14），結合（式 3-6）與（式 3-12）可得（式 3-15），根據 Gradient Method，$\boldsymbol{\epsilon}^{(k)}$可求出如（式 3-16）。

$$R_{ij}^{(k)} = \frac{1}{m_s n_s}\sum_{m=1}^{m_s}\sum_{n=1}^{n_s}\left|\mathcal{H}_i^{(k)}(m,n)\right|\mathrm{sgn}\left(\mathcal{H}_i^{(k)}(m,n)\right)\mathcal{H}_j^{(k)}(m,n) \tag{3-13}$$

$$\mathbf{D}^{(k)} = \left[\mathrm{diag}\left(R_{11}^{(k)},\ldots,R_{NN}^{(k)}\right)\right]^{-\frac{1}{2}} \tag{3-14}$$

$$G\left(\boldsymbol{\mathcal{H}}^{(k+1)}\right) = G\left(\boldsymbol{\mathcal{H}}^{(k)}\right) + \mathrm{Tr}\left(\boldsymbol{\epsilon}^{(k)}\mathbf{R}^{(k)T}\mathbf{D}^{(k)}\right) \tag{3-15}$$

$$\boldsymbol{\epsilon}^{(k)} = -\mu\mathbf{D}^{(k)}\mathbf{R}^{(k)} \tag{3-16}$$

使用$\boldsymbol{\epsilon}^{(k)}$來更新分離矩陣 **B**，直至$\left\|\mathbf{B}^{(k+1)} - \mathbf{B}^{(k)}\right\| < \delta$，$\delta$為一個很小的閾值，表示 **B** 已收斂，將 **B** 乘上白化矩陣即可做為影像的分離矩陣，將此矩陣與原混和影像 **X**(m,n) 相乘，就可得到原影像 **S**(m,n)。

3.4 Initial Condition

針對這種用疊代法求最佳解的問題，初始條件通常是很重要的，在此我們用 FastICA 的解當作初始條件，如此一來可以降低疊代的次數，且 FastICA 的收斂速度本身就很快，因此並不會增加演算法的複雜度，整個流程圖如下：

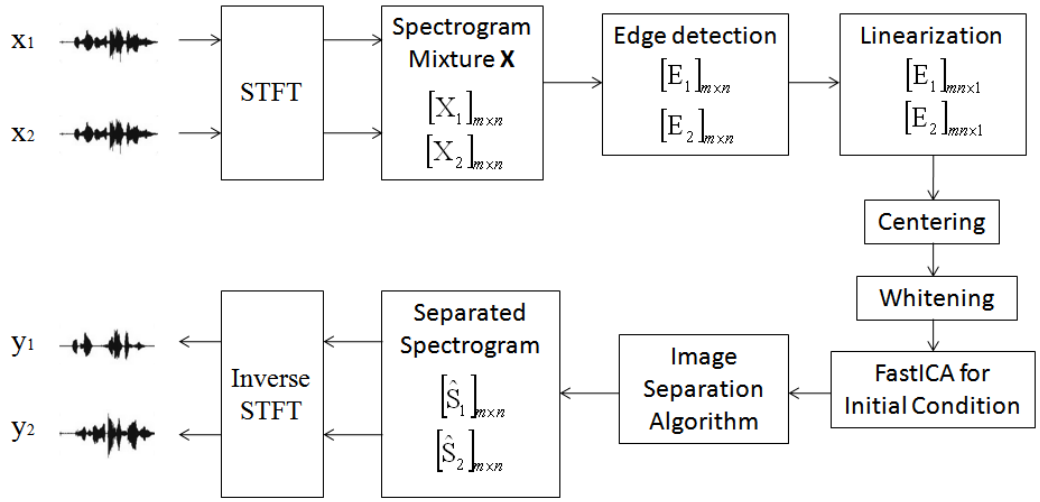
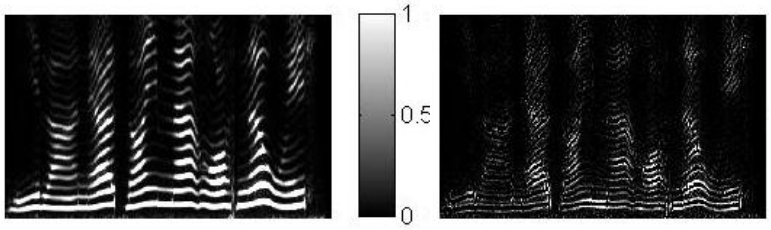


圖 3-2　FastICA 與 ISBS 流程圖

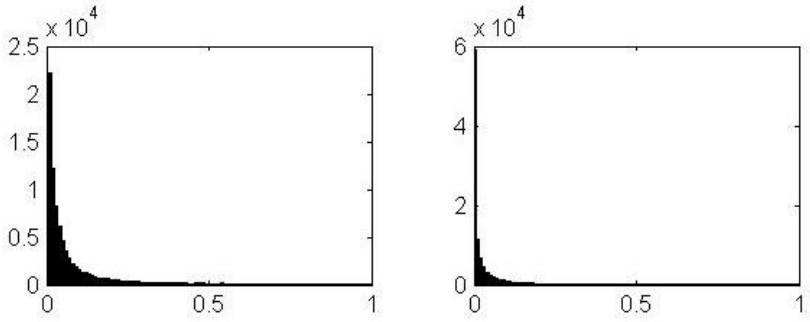
3.5 影像分離演算法用於聲譜圖分離

從我們初步的實驗中發現，影像分離演算法在訊號源來自於頭的不同邊時，經過 HRTF 模擬混和的訊號尚可分離，但對於訊號源來自同邊的混和訊號，解出來的效果有限，第五章中會再顯示詳細的模擬結果。

討論其原因，此方法的概念是將聲譜圖作為影像訊號來處理，希望利用邊界圖會比原影像訊號來得稀疏的特性來做分離，但結果顯示聲譜圖與影像訊號仍然有不同的特性，聲譜圖的邊界圖不會如影像訊號的邊界圖那麼的稀疏。（圖 3-3）為一聲譜圖與其邊界圖的統計直方圖，比較圖(d)與圖(c)，可看出邊界圖確實會比原影像圖來得稀疏，但其差異就不及（圖 3-1）來得明顯，這也可能是影像分離演算法對聲譜圖無法正確分離的原因所在。



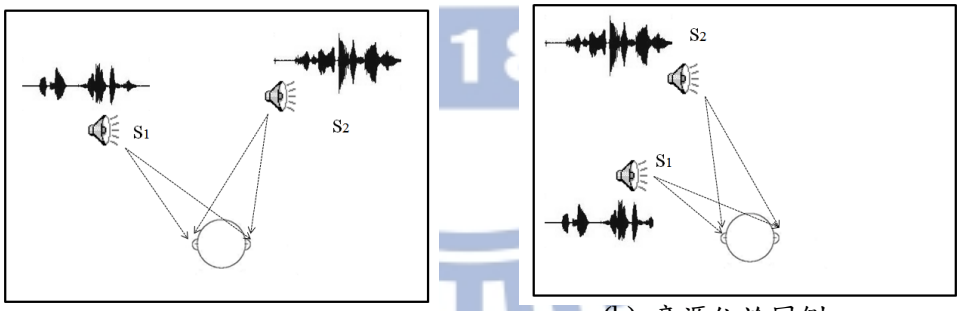
(a)原聲譜圖　　(b)聲譜圖之邊界圖



(c)原聲譜圖之統計直方圖　　(d)邊界圖之統計直方圖

圖 3-3　聲譜圖與邊界圖及其統計直方圖

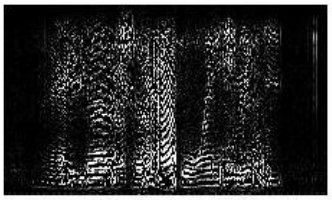
在音源方位不同的混和示意圖如（圖 3-4）所示，圖(a)為兩個音源位於頭的不同兩側，圖(b)為兩個音源於頭的同一側，在這兩種情況下，邊界圖的稀疏情況也會有所不同。



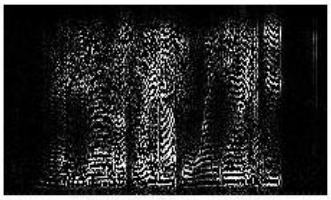
(a) 音源位於不同側　　(b) 音源位於同側

圖 3-4　混和示意圖

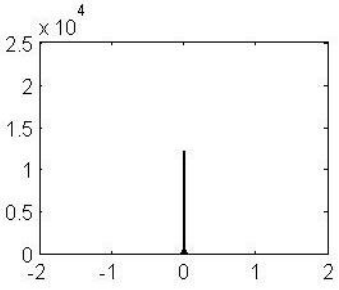
（圖 3-5）為音源位於不同側時，兩耳所收到的混和訊號 x_1 與 x_2，其聲譜圖的邊界圖 E_1 與 E_2，以及邊界圖的統計直方圖；（圖 3-6）為音源位於同側時的情況。比較（圖 3-5）與（圖 3-6）的圖(c)與圖(d)，可看出當音源位於不同側時，E_1 與 E_2 的稀疏性會被降低，因此以提高稀疏性作為分離目標的演算法是可行的；而當音源位於同側時，其中一個邊界圖 E_1 本身就較為稀疏，因此在這種情況下演算法較容易失敗。



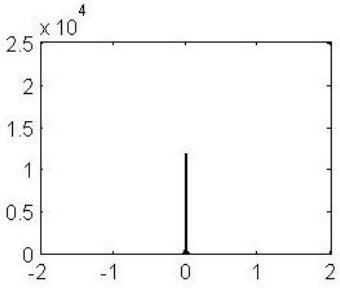
(a) x_1 之聲譜圖邊界圖 E_1



(b) x_2 之聲譜圖邊界圖 E_2

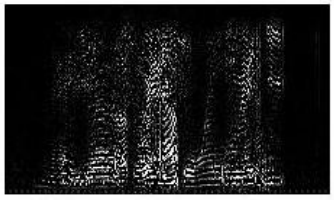


(c) E_1 之統計直方圖

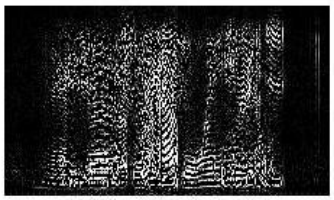


(d) E_2 之統計直方圖

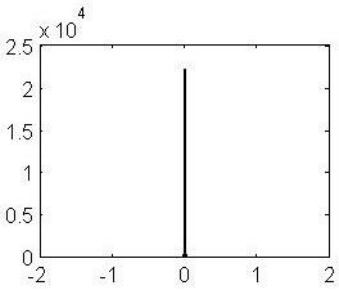
圖 3-5　音源位於不同側的混和訊號



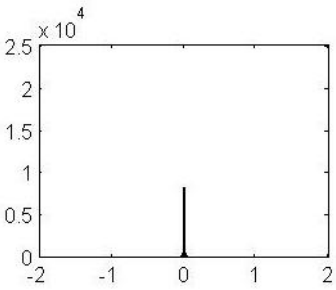
(a) x_1 之聲譜圖邊界圖 E_1



(b) x_2 之聲譜圖邊界圖 E_2



(c) E_1 之統計直方圖



(d) E_2 之統計直方圖

圖 3-6　音源位於同側的混和訊號

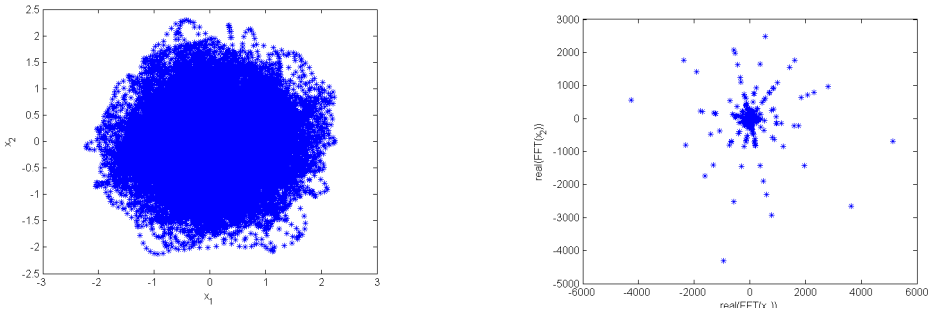
第四章　稀疏成分分析法

從我們初步的實驗中發現，影像分離演算法在訊號源來自於頭的不同邊時，經過 HRTF 模擬混和的訊號尚可分離，但對於訊號源來自同邊的混和訊號，解出來的效果有限。因此除了將整張聲譜圖當成影像訊號來做處理外，也該考慮聲譜圖本身所具備的特性，若能將此特性加入分離的考量，應該會有比較好的效果。

聲譜圖可視為語音頻率成分隨時間的變化，而不同的人講話會有不同的基頻與倍頻，說話速度與斷句也不一，所以不同音源的聲譜圖的交集是很少的（disjoint），因此混和訊號聲譜圖的每個 T-F unit 都只來自於其中一個訊號源，也就是有很稀疏（sparse）的特性[20]，近年來所發展的稀疏成分分析法（Sparse Component Analysis, SCA），就是適用於此種訊號分離。

4.1 發展背景

如同緒論中所提到的，在盲訊號分離法中，對於訊號源不同的假設會產生不同的演算法，有別於 ICA 利用訊號獨立的統計特性作為基本假設，近年來開始有人提出一個新的假設，也就是假設訊號源在某些 domain 上是非常稀疏的，形成了所謂的稀疏成分分析法[17-20]。如果一個訊號多數的值都為 0 或是趨近於 0，就稱為稀疏訊號（sparse signal），其訊號分佈函數在 0 附近會有一個高峰（peak），其餘地方值都很小。將語音訊號轉到 sparse domain 的方法多為 Fourier Transform 或是 Wavelet Transform（圖 4-1），圖(a)為 3 個訊號源經過即時性混和成 2 個混和訊號，x_1 與 x_2 在時間軸的散佈圖，圖(b)為經過 512-point windowed FFT 後的係數絕對值之散佈圖。當訊號轉到 sparse domain 後，兩個以上的訊號源同時出現的機率會很低，也就是在同一時間點上大部分只有一個訊號源是有值的，也就形成了很好的分離條件。

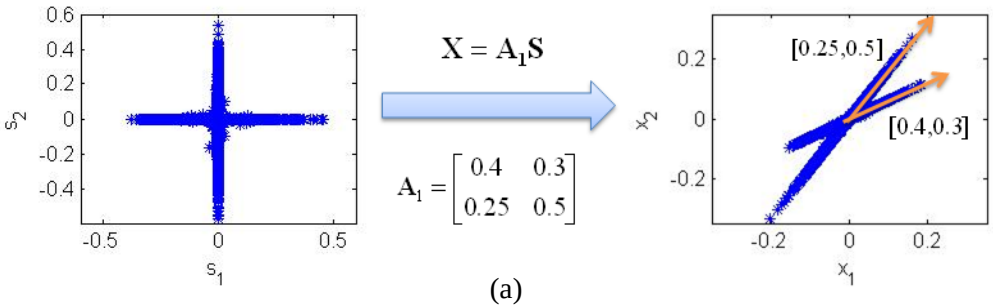


(a) 時域混和訊號的散佈圖　　(b) 經過 FFT 後的散佈圖

圖 4-1 訊號在時域與頻域的散佈情況

4.1.1 針對即時性混和的 SCA

針對即時性混和的 SCA 主要就是在訊號的散佈圖上找方向[17]，因為對即時性混和訊號來說，混和訊號的散佈圖方向就代表混和矩陣 **A** 的列向量，（圖 4-2）以兩個很稀疏的語音訊號為例，(a)圖左方為兩個訊號源的散佈圖，右方為經過矩陣 $\mathbf{A}_1$ 混和後，兩個混和訊號的散佈圖，可看出散佈圖上的兩個主要方向即為 $\mathbf{A}_1$ 的列向量；(b)圖為訊號源經過另一個混和矩陣 $\mathbf{A}_2$ 的情況，同樣也是在混和訊號的散佈圖中可以找出 $\mathbf{A}_2$ 的方向。因此找出這樣的方向變化便可以用來預估混和矩陣，如 M. Zibulevsky 於 2002 年提出以 k 均值分群法（k-means clustering）來歸類主要方向[19]；另外，因為此想法也就近似於找出變異數（variance）最大的方向，因此對混和訊號的共變異數矩陣做特徵值分解，其最大值的特徵值所對應的特徵向量，也可看作是散佈圖上的主要方向，這樣的方法也就是主成分分析（Principle Component Analysis, PCA）的基本原理，PCA 通常在 ICA 的演算法內會作為 pre-whitening 的前置處理，在第二章內有詳細的介紹。



(a)

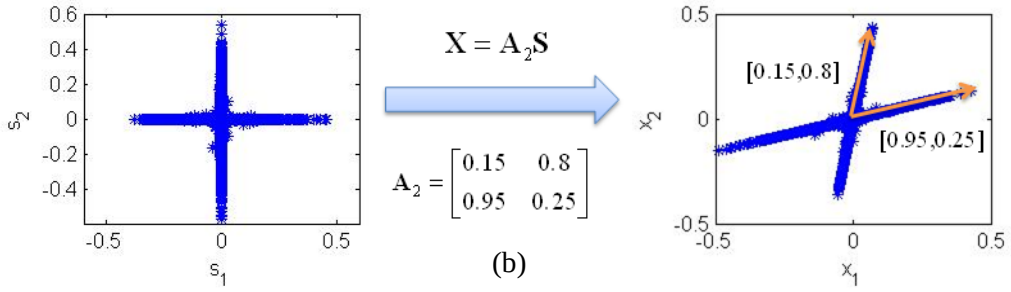


圖 4-2　訊號源經過即時性混和後方向的變化

4.1.2　DUET

針對頻域訊號的 SCA，最著名的就是 S. Rickard 與 O. Yilmaz 於 2000 年提出的 DUET（Degenerate Unmixing and Estimation Technique）[20]，其做法也是將訊號利用 STFT 轉到頻域，利用頻域訊號較時域訊號更為稀疏的特性來做分離。由（圖 4-3）可看出訊號源的聲譜圖是非常稀疏的，(a)圖為訊號源 s_1 的聲譜圖 $|S_1|$，(b)圖為 s_2 的聲譜圖 $|S_2|$，(c)圖為$|S_1|$與$|S_2|$點對點相乘，可以看出相乘後大部分的值都變得很小，表示$|S_1|$和$|S_2|$同時出現的時間點是很少的，(d)圖為(c)圖的統計直方圖，圖中也顯示大部分的值都集中在 0 附近。

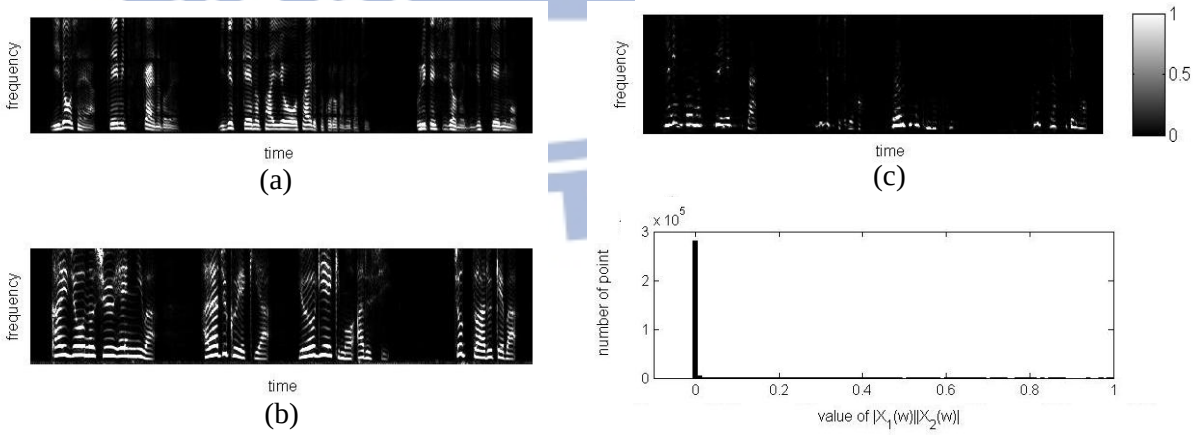


圖 4-3　聲譜圖的稀疏特性

DUET 是假設訊號經過不同的延遲（delay）和衰減所混和而成的（式 4-1），可以看作是摺積混和訊號中，只考慮由直接路徑（direct path）傳到麥克風的音源，不考慮音源

經過其他反射後的混和情況。若只考慮 x2 相對 x1 的延遲和衰減情況，（式 4-1）可簡化為（式 4-2），經過 STFT 轉換後可寫成（式 4-3）。

$$x_1(t) = \sum_{i=1}^{n} a_{1i} s_i(t - \delta_{1i})$$
$$x_2(t) = \sum_{i=1}^{n} a_{2i} s_i(t - \delta_{2i}) \tag{4-1}$$

$$x_1(t) = \sum_{i=1}^{n} s_i(t)$$
$$x_2(t) = \sum_{i=1}^{n} a_i s_i(t - \delta_i) \tag{4-2}$$

$$\begin{bmatrix} X_1(\omega,\tau) \\ X_2(\omega,\tau) \end{bmatrix} = \begin{bmatrix} 1 & \dots & 1 \\ a_1 e^{-j\omega\delta_1} & \dots & a_n e^{-j\omega\delta_n} \end{bmatrix} \begin{bmatrix} S_1(\omega,\tau) \\ \vdots \\ S_n(\omega,\tau) \end{bmatrix} \tag{4-3}$$

假設每個 T-F unit 都只有來自其中一個音源 s_i 的訊號，表示 X1(ω,τ)和 X2(ω,τ)都只包含音源 $S_i(\omega,\tau)$（式 4-4），因此只需利用混和訊號的兩張聲譜圖在當點(ω,τ)的振幅及相位差，就可以計算音源 s_i 的混和係數(a_i,δ_i)，如（式 4-5）所式。

$$\begin{bmatrix} X_1(\omega,\tau) \\ X_2(\omega,\tau) \end{bmatrix} = \begin{bmatrix} 1 \\ a_i e^{-j\omega\delta_i} \end{bmatrix} S_i(\omega,\tau) \tag{4-4}$$

$$(a_i, \delta_i) = \left(\frac{|X_2(\omega,\tau)|}{|X_1(\omega,\tau)|}, \Im\left\{ \log\left(\frac{X_2(\omega,\tau)}{X_1(\omega,\tau)} \right) \right\} / \omega \right) \tag{4-5}$$

針對每個(ω,τ)都可以算出一組混和係數(a,δ)，而來自同一個音源的混和係數假設上都是一樣的，因此若將這些係數作分群，就可以得到音源訊號的混和係數，具體的作法是將這些係數畫出其統計直方圖，其中每個高峰就會對應出一個音源（圖 4-4）。最後只要將屬於此音源的 T-F unit 保留，遮蔽掉不屬於該音源的 T-F unit，會得到一個遮蔽矩陣（masking matrix）M_i（式 4-6），將此遮蔽矩陣與混和訊號的聲譜圖相乘就可以得到訊號源 s_i 的聲譜圖（式 4-7），再用 inverse STFT 求得原本的聲音源。

$$M_i(\omega,\tau) = \begin{Bmatrix} 1 & \text{if}(\omega,\tau) \in \text{source i} \\ 0 & \text{otherwise} \end{Bmatrix} \tag{4-6}$$

$$S_i(\omega,\tau) = M_i(\omega,\tau) X_1(\omega,\tau) \tag{4-7}$$

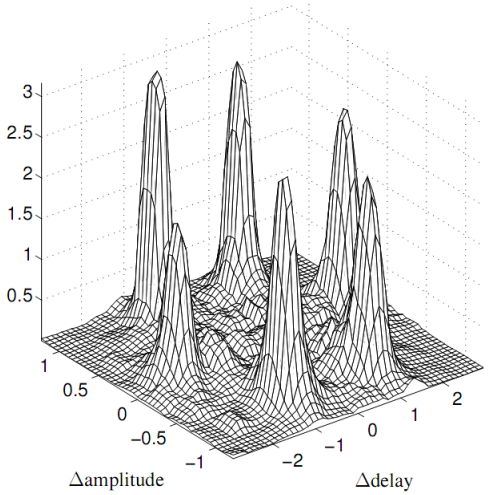
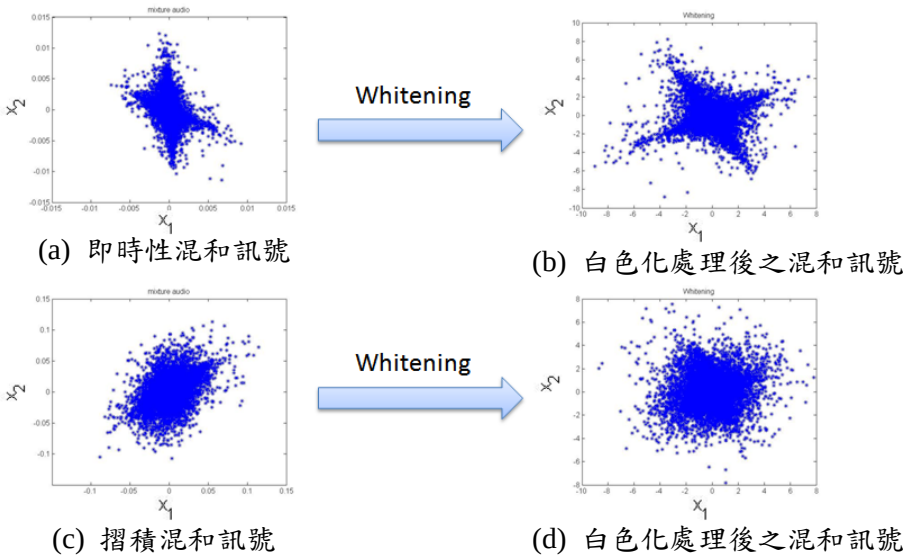


圖 4-4　振幅和相位差之統計直方圖

4.2 稀疏訊號轉換

我們原先是將聲譜圖從影像訊號的觀點切入，但從模擬結果中發現，影像分離演算法對於訊號源來自同邊的混和訊號，解出來的效果有限，探討其可能的原因，首先從摺積混和訊號和即時性混和訊號的語音訊號開始說明。在即時性混和訊號中，每個點加乘的權重（weight）都是一樣的，因此在散佈圖上會呈現出兩個主要的方向（圖 4-5(a)），如此在做白化處理後才能確實達到 de-correlation 的效果（圖 4-5(b)）；而摺積混和訊號的散佈圖中無法看出有兩個主要的方向（圖 4-5(c)），就算做了白化處理也沒辦法 de-correlation（圖 4-5(d)）。

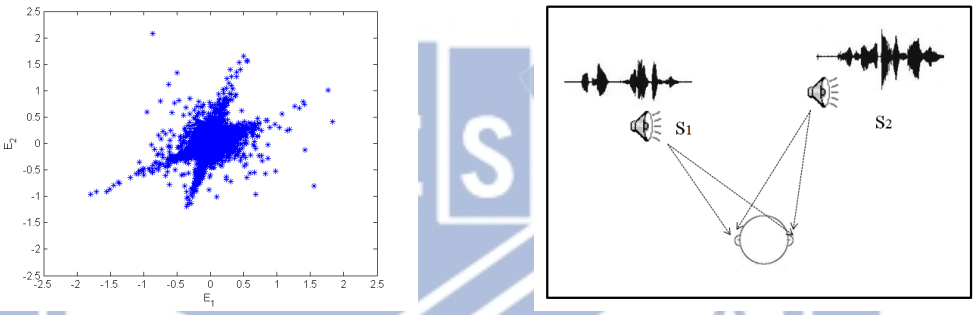


(a) 即時性混和訊號　(b) 白色化處理後之混和訊號

(c) 摺積混和訊號　(d) 白色化處理後之混和訊號

圖 4-5　混和訊號及經過白色化處理後的變化

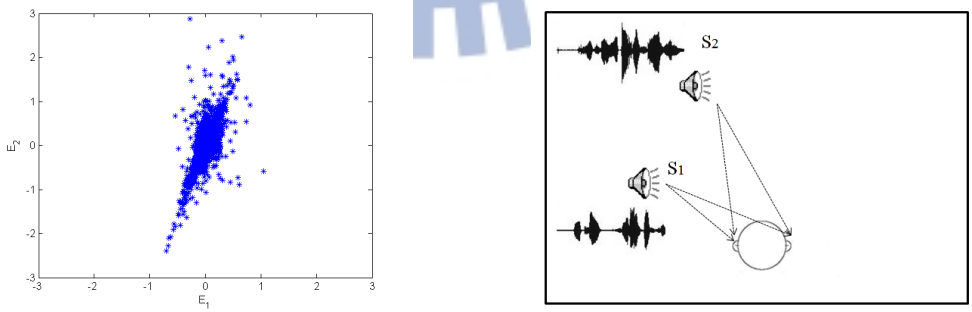
訊號在做摺積混和時，若是訊號源到麥克風的反射路徑並不多，HRTF 的脈衝響應長度就不會很長，如果其長度是在 STFT 的音框長度(frame size)及音框位移(frame shift) 足以覆蓋的長度以內的話，摺積混和訊號在經過 STFT 之後，便可以近似於即時性混和訊號[25]；(圖 4-6(a)) 為脈衝響應長度較短時，對聲譜圖的邊緣所做的散佈圖，可看出其分佈近似於即時性混和訊號。但是散佈圖的方向其實主要來自於混和訊號 1 與混和訊號 2 各點數值大小的差異，當兩個聲源訊號分別位於頭的兩側時（圖 4-6(b)），音源 1 在混和訊號 1 比較大聲，在混和訊號 2 比較小聲，而音源 2 則是在混和訊號 2 比較大聲，在混和訊號 1 比較小聲，如此一來混和訊號 1 和混和訊號 2 在在每個時間點上都會有不同的主要音源，所以散佈圖上的方向也會較為明顯。



(a) 聲譜圖邊緣之散佈圖　　(b) 混和示意圖

圖 4-6　訊號源位於不同邊的混和情形

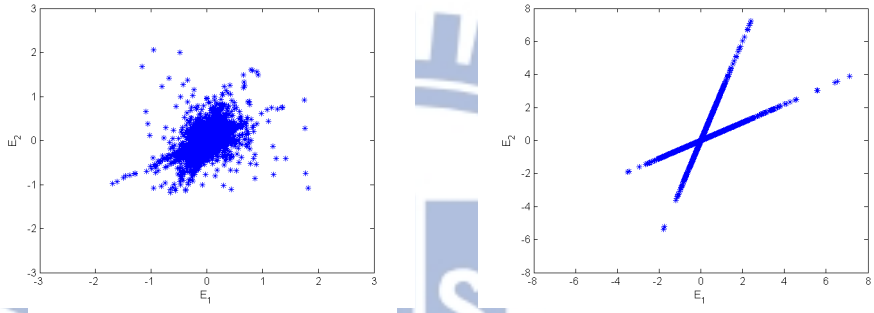
反觀兩個音源訊號在同一邊的情況（圖 4-7(b)），音源 1 和音源 2 都是在混和訊號 1 比較大聲，在混和訊號 2 比較小聲，也就是兩個音源主要都集中在其中一個混和訊號，因此散佈圖上就不容易找出另一個方向（圖 4-7(a)）。



(a) 聲譜圖邊緣之散佈圖　　(b) 混和示意圖

圖 4-7　訊號源位於同邊的混和情形

當摺積混和訊號的脈衝響應越來越長時，在 STFT domain 產生類似即時性混和的特性也會漸漸消失，因此聲譜圖邊緣的散佈圖也會變得難以辨認方向（圖 4-8(a)），在此情況下就如同前段所述，白化處理無法達到 de-correlation 的目的。因此我們希望可以找到一種投影的方法，將分散的點集中到兩個主要的方向上（圖 4-8(b)），如此在經過白化處理後可以得到兩個非相關的訊號，希望可以達到較好的分離效果。這樣的想法就相當於將訊號轉到一個稀疏的領域，用此稀疏訊號來做分離。



(a) 聲譜圖邊緣之散佈圖　　(b) 投影成稀疏訊號

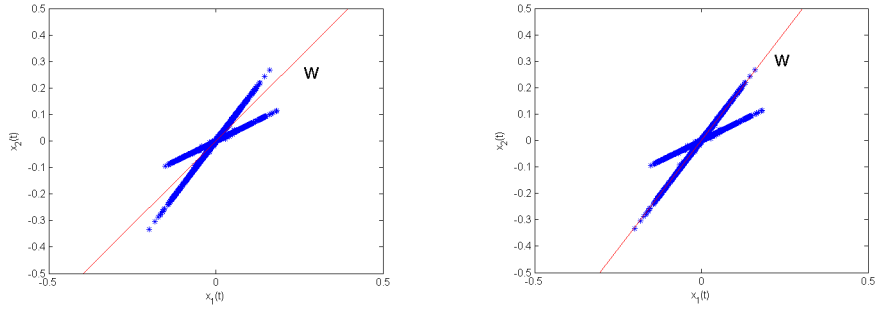
圖 4-8　長脈衝響應下的混和情況

4.3 Nonlinear Projection Column Masking

為了將訊號投影到稀疏領域，首先要找出其主要的兩個方向。如（圖 4-9）所示，假設我們想要找到的方向向量為 **w**，則散佈圖上各點對於 **w** 的投影量 y_t 定義如下：

$$y_t = \|x_t\| * |\cos(\widehat{\mathbf{w}, \mathbf{x}_t})| \tag{4-8}$$

其中 $\mathbf{x}_t$ 為各點的座標向量，$\widehat{\mathbf{w}, \mathbf{x}_t}$ 表示兩個向量 **w** 與 $\mathbf{x}_t$ 的夾角。PCA 是對於各個方向的 **w**，找出使 $E[y_t^2]$最大化的方向向量 **w**，E[.]代表期望值。如（圖 4-9）所示，此方法在預估 **w** 時，會受到另一個方向的訊號點所干擾，雖然當 $\mathbf{x}_t$ 與 **w** 的夾角很大時，$|\cos(\widehat{\mathbf{w}, \mathbf{x}_t})|$的值會比較小，但全部的點所加總起來的影響，還是會使求出的 **w** 稍微偏離真正所要找的方向（圖 4-9(a)）。



(a) PCA (b) NPCM

圖 4-9 散佈圖上的主要方向

為避免此問題，我們可在找方向時，先遮蔽掉遠離 **w** 的所有訊號點，如此一來就較不會受到屬於另一個方向的訊號點所干擾，可以較正確地找出 **w** 的方向（圖(b)）。[26]提出了一個非線性投影的方法將遠離主要方向的點遮蔽掉（Nonlinear Projection Column Masking, NPCM），其想法是定義出一個閾值 c_0（threshold），對於每個點 x_t 計算 $|\cos(\widehat{\mathbf{w},\mathbf{x}_t})|$，當 $|\cos(\widehat{\mathbf{w},\mathbf{x}_t})| \geq c_0$ 時表示 x_t 與 **w** 較為接近，再對這些點做 PCA。但是在這樣的定義下，閾值 c_0 的選擇對效果好壞會有很大的影響，所以文中提出了一個非線性函數 f(.)，將投影量 y_t 重新定義為：

$$y_t = \|x_t\| * f(\cos(\widehat{\mathbf{w},\mathbf{x}_t})) \tag{4-9}$$

此 f(.)為一遞減函數，並滿足以下三個條件：

(1) $0 \leq f(\alpha) \leq 1$
(2) $f(-\alpha) = f(\alpha)$
(3) $f_{max} = f(\pm 1)$ and $f(\alpha) \ll f_{max}$ if $|\alpha| \leq c_0$

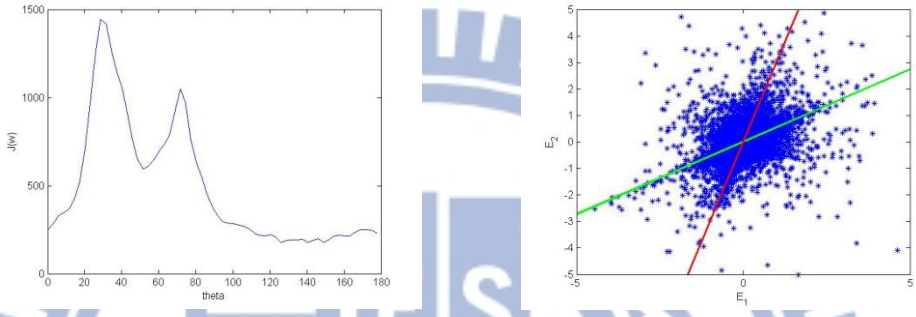
條件(3)就表示當 x_t 與 **w** 在同一個方向上時，f(.)會有最大值，而遠離 **w** 的點對於 y_t 的貢獻值就很小，如此可以達到遮蔽的效果。f(.)可以是任意滿足條件的函數，在此使用 $f(x) = \exp(-\rho x^2)$。其中 ρ 為常數，根據其值大小可以決定指數函數 f(x)的衰減速度，若 ρ 越大，f(x)衰減越快，表示夾角 $\widehat{\mathbf{w},\mathbf{x}_t}$ 一定要很小，x_t 才不會被遮蔽，也就是 ρ 值可以控制訊號點被遮蔽的門檻。

根據以上的假設，可以定義出目標函數：

$$\max_{\mathbf{w}} J(\mathbf{w}) = \sum_t \|\mathbf{x_t}\| f(\cos(\widehat{\mathbf{w},\mathbf{x}_t})) \tag{4-10}$$

對各個不同方向的 $\mathbf{w}$ 計算 J(w)值，當 J($\hat{\mathbf{w}}$)有最大值時，$\hat{\mathbf{w}}$ 就應該是我們要求的主要方向。假設有兩個聲源訊號，J($\mathbf{w}$)就會有兩個高峰（圖 4-10(a)），這兩個高峰也就對應了主要的兩個方向 $\mathbf{w}_1$、$\mathbf{w}_2$ 的角度（圖 4-10(b)）。而根據當點 x_t 所計算出的函數值 $f(\cos(\widehat{\mathbf{w}_k, \mathbf{x}_t}))$的大小，可以決定 x_t 距離哪個方向比較近，再將此點投影到該方向上（式 4-11）。

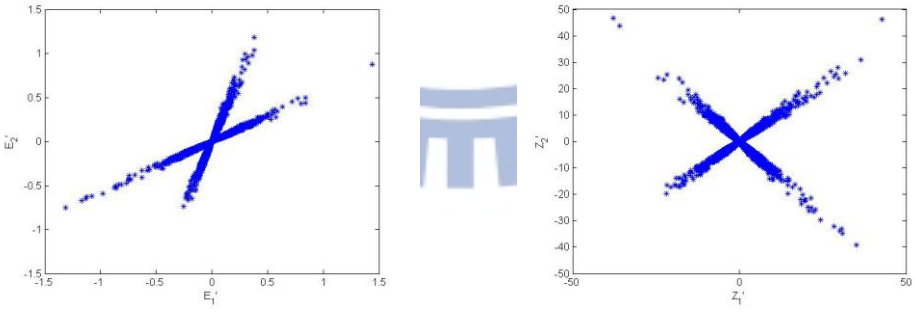
$$x_t' = \|x_t\| * f(\cos(\widehat{\mathbf{w}_k, \mathbf{x}_t})) \tag{4-11}$$



(a) (b)

圖 4-10　NPCM 所找出的兩個方向

對每個點 x_t 皆投影到 $\mathbf{w}_1$ 或 $\mathbf{w}_2$ 的方向上後，就可以得到一個方向很明確的散佈圖（圖 4-11(a)），此經過投影的訊號 x_t' 經過白化處理後，就可以達到 de-correlation 的效果（圖 4-11(b)），然後再繼續解分離矩陣。



(a) (b)

圖 4-11　稀疏訊號經過白色化處理後的變化

將投影過的訊號當作新的混和訊號 x'，用 x'所解出的白化矩陣和分離矩陣再回來解 x，流程圖如（圖 4-12）所示，我們將原本聲譜圖的邊緣訊號 E_1 和 E_2 作 NPCM，會得

到一個新的訊號 E_1'和 E_2'，用 E_1'、E_2'所解出的白化矩陣與分離矩陣乘上原影像 $\mathbf{X}_1$、$\mathbf{X}_2$，就可以得到分離出的聲譜圖 $\widehat{S_1}$、$\widehat{S_2}$。

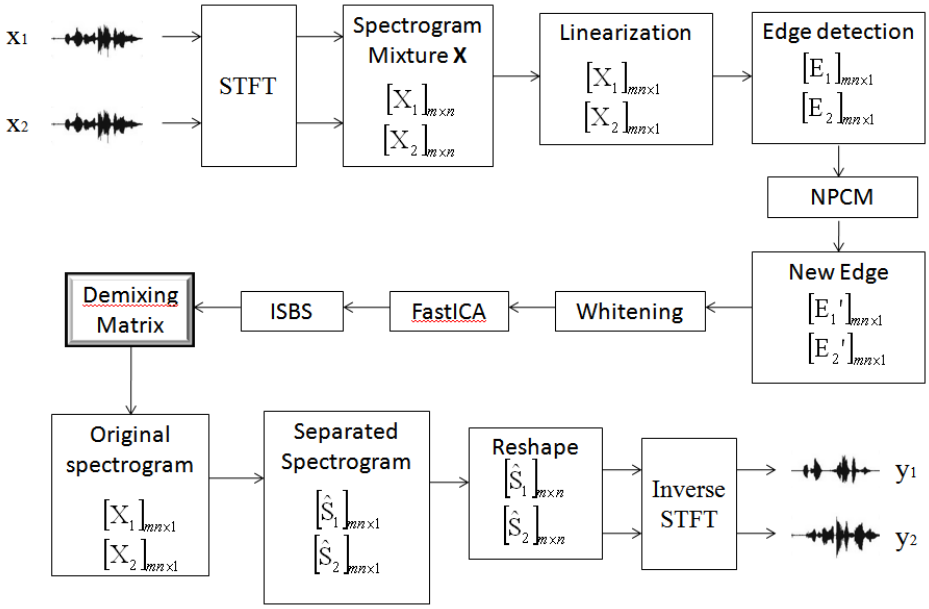


圖 4-12　加入非線性投影的影像分離法

此方法是將一個新訊號 x′的解直接用在原訊號 x 上，但因為 x 到 x′之間的轉換為非線性的，將 x 轉換成 x′的過程就破壞了訊號原本的特性，所以也是只有在訊號源位於不同邊的情況下效果比較好，若是訊號源來自同一邊，還是會因為混和訊號太過相似，導致分離效果依然有限，詳細的結果會在第五章再作討論。

4.4 Proposed Algorithm

前面所述的方法都是希望可以直接找出一個分離矩陣，乘上混和訊號後就可以解出原本的訊號。但對摺積混和訊號而言，本身它的混和過程就不是矩陣的相乘，原本希望轉到影像維度上會比較近似於兩張影像的線性混和，但對於音源來自同邊的混和而言，因為兩個混和訊號過於類似，所以不太可能用加乘的方法直接解出來。

為解決摺積混和訊號之分離，我們參考 Convolutive ICA 的做法，將訊號聲譜圖分頻，再將每個頻率的訊號當作一組混和訊號，並加入聲譜圖為稀疏訊號的假設來做分離，個別頻率的訊號都分離完之後，再用 overlap and add 的方法將聲譜圖還原成聲音。

首先，我們將聲音訊號做 STFT（式 4-12），把語音訊號轉成聲譜圖（圖 4-13），將每一個頻率 ω_k 的訊號 $\mathbf{X}(\omega_k,\tau)=[X_1(\omega_k,\tau), X_2(\omega_k,\tau)]$ 當作一組混和訊號，在只有兩個音源的情況下，其散佈圖會有兩個主要的方向（圖 4-14），集中在 $\mathbf{w}_1$ 附近的點表示在混和訊號 1 較大聲，在混和訊號 2 裡較小聲，反之，集中在 $\mathbf{w}_2$ 附近的點表示在混和訊號 2 較大聲，在混和訊號 1 裡較小聲。根據稀疏訊號的假設，$\mathbf{X}(\omega,\tau)$ 的每個 T-F unit 都只有來自於其中一個音源，因此我們可以假設 $\mathbf{w}_1$ 附近的點就代表音源 1，在 $\mathbf{w}_2$ 附近的點就代表音源 2。

$$\mathbf{X}(\omega,\tau) = \sum_{\tau} x(t)w(t-\tau)e^{-j\omega\tau} \tag{4-12}$$

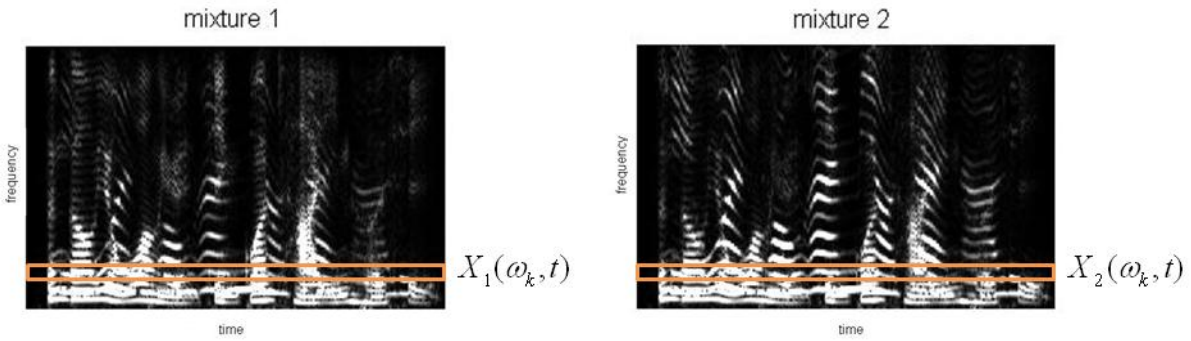


圖 4-13　混和訊號聲譜圖分頻

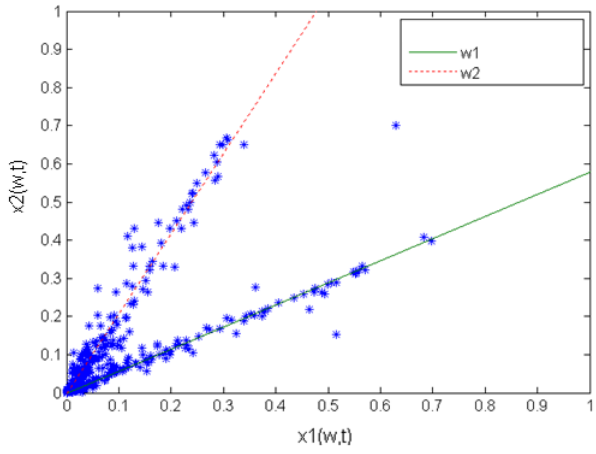


圖 4-14　某頻帶的混和訊號分佈圖

在此假設之下，如果想要分離出音源 1，我們就必須將 $\mathbf{w}_1$ 附近的點放大，並同時遮蔽 $\mathbf{w}_2$ 附近的點，這些遮蔽值的大小也可以利用先前的非線性函數來計算（式 4-13）：

$$u_j(\omega,\tau) = \exp\left(-\kappa \cos^2\left(\mathbf{w}_j, \widehat{\mathbf{X}(\omega_k,\tau)}\right)\right) \quad (4\text{-}13)$$

由（式 4-13）可以計算出各點對於 $\mathbf{w}_1$ 的貢獻值，其中 κ 為一介於 10^4~10^7 的常數，根據其值大小可以決定指數函數的衰減速度，若 κ 越大，u(t)衰減越快，表示訊號點 x_t 只有在距離方向向量 $\mathbf{w}$ 很近時，才會有比較大的遮蔽值，因此 κ 值可以控制訊號點被遮蔽的門檻。若兩個音源的來源方向相差較大，遮蔽門檻不需要太嚴格就可以分離訊號，κ 值也就不用太大；若兩個音源的來源方向較近，就需要比較嚴格的遮蔽門檻，就要設定較大的 κ 值。

最後只要將此遮蔽函數$u_j(t)$與混和訊號 1 和混和訊號 2 相乘，就可以解出原本的音源（式 4-14），其中・表示點對點相乘。

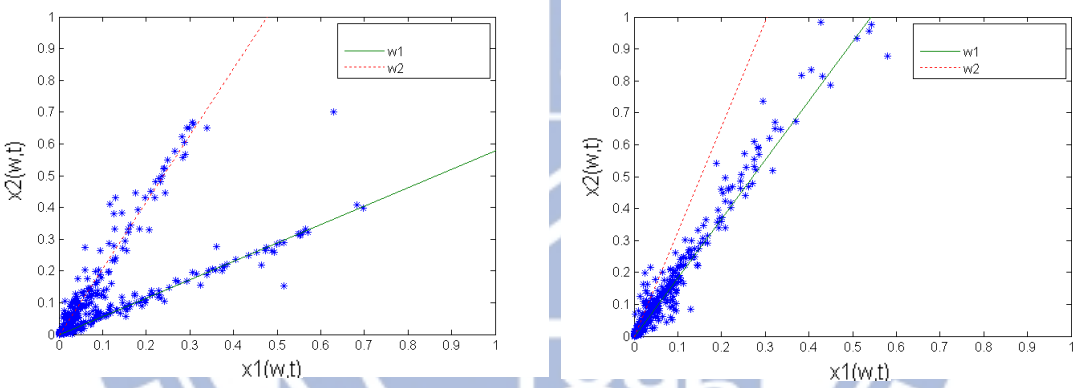
$$S_j(\omega_k,\tau) = \sum_{i=1}^{2} u_j(\omega,\tau) \cdot X_i(\omega_k,\tau) \quad (4\text{-}14)$$

4.5 排列問題（Permutation）

正如 Convolutive ICA 一樣，我們將每個頻帶的訊號當作一組混和訊號獨立來解，每個 ω_k 所解回來的$\hat{s_1}(\omega_k,\tau)$和$\hat{s_2}(\omega_k,\tau)$的順序是不太一定的（圖 1-4），因此解決此排列問題是一個很重大的議題。目前一種比較常見的方法是用訊號來源方向（Direction of Arrival, DOA Approach）來分類訊號，利用聲源從不同的方向抵達會有不同的延遲，解出來的分離矩陣 W(ω)也會有不同的相位差，利用這些相位差將每個頻帶所解出的訊號分群，最後歸在同一群的就應該是同一個訊號源。另外，因為聲譜圖具連續性，也有方法是利用相鄰頻帶的訊號會有比較高的關聯性，用該頻帶訊號與相鄰頻帶訊號的相關性（correlation）大小作為排列依據（Correlation Approach）。用相關性排序的方法較為直覺，準確性也較高，因為 DOA 的判斷還要取決於分離矩陣的正確性；然而 DOA Approach

會比 Correlation Approach 還要穩健（robust），因為在 Correlation Approach 裡若有其中一個頻帶判斷錯誤，就很容易造成其他頻帶也出錯。[16]

在我們所提出的演算法中，排列問題變得相對容易解決，因為在運算過程中所找出的兩個方向，其實就分別是這兩個音源對混和訊號的貢獻，雖然每個頻帶所找出的方向並不會完全一樣，但當兩個音源的來源角度相差較遠，或是分別來自左右兩方時，因為音源對於混和訊號的貢獻程度相差較大，所以 $\mathbf{w}_1$ 和 $\mathbf{w}_2$ 的角度也會有比較大的差別（圖 4-15(a)）。而集中在 $\mathbf{w}_1$ 和 $\mathbf{w}_2$ 附近的點又分別代表音源 1 和音源 2，所以這時只要將每個頻帶所解出來的 $\mathbf{w}_1$ 和 $\mathbf{w}_2$ 依角度大小排列，就可以確定 $\mathbf{w}_1$ 所解出的是音源 1，$\mathbf{w}_2$ 所解出的是音源 2，如此一來就可以解決排列問題。



(a) 訊號源相距較遠　　(b) 訊號源相距較近

圖 4-15　訊號源方向對混和訊號散佈圖的影響

然而當兩個音源的來源方向太近，兩個音源對混和訊號的貢獻程度相差不大（圖 4-15(b)）；或是有多個音源的時候，來自不同音源的貢獻值較容易混淆，此時如果只用 w 的角度來排序，還是有可能會有排列錯誤的問題。在此我們利用聲譜圖的連續性作為判斷的準則，也就是計算該頻帶所解出的訊號和附近頻帶所解出訊號的差異大小來判斷（式）。如果 C_i 的值最小，表示此訊號是與音源 i 的頻譜圖最相像，那此訊號就應該是屬於音源 i 的。大部分的比較方法是與前一個頻帶的訊號做比較（式 4-15），但因為語音訊號在 harmonic 之間的空檔有時會造成誤判，所以在（式 4-16）中多設了一個 α 值，也就是可以跟前 α 個頻帶的訊號比較，α 值若是可以配合訊號的 harmonic，判斷的準確

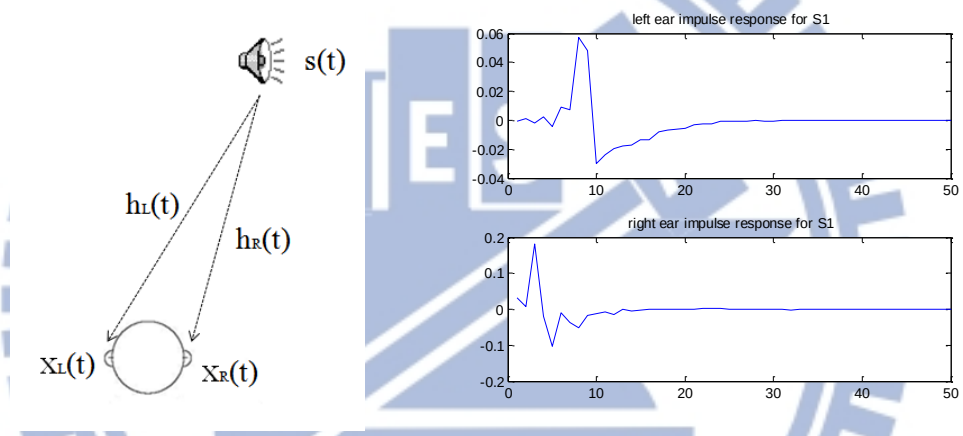
度也會提高。

$$C_i = \sum_t [\hat{s}(\omega_k, t) - \hat{s}_i(\omega_{k-1}, t)]^2 \tag{4-15}$$

$$C_i = \sum_t \{ \hat{s}(\omega_k, t) - [\sum_{\ell=1}^{\alpha} \hat{s}_i(\omega_{k-\ell}, t)]/\alpha \}^2 \tag{4-16}$$

第五章　模擬結果

我們使用頭部相關傳輸函數（Head-Related Transfer Function, HRTF）作為摺積混和的脈衝響應，也就是模擬當聲音源來自某個方位角時，雙耳所聽到的聲音。HRTF 包含聲源到左耳的反射路徑的脈衝響應 $h_L(t)$，及到右耳的脈衝響應 $h_R(t)$（圖 5-1），當音源訊號與 $h_L(t)$和 $h_R(t)$做摺積之後，便會在左右耳產生不同的時間延遲與強度，也就分別模擬左耳和右耳所收到的聲音。



(a) 示意圖　　(b) 音源到左右耳的脈衝響應

圖 5-1　HRTF 混和訊號

當聲源位於不同的位置時，對於左右耳會有不同的脈衝響應，因此我們的混和訊號可寫成（式 5-1）：

$$x_L(t) = h_{1L}(t) * s_1(t) + \cdots + h_{iL}(t) * s_i(t) = \sum_{i=1}^{n} h_{iL} * s_i(t)$$
$$x_R(t) = h_{1R}(t) * s_1(t) + \cdots + h_{iR}(t) * s_i(t) = \sum_{i=1}^{n} h_{iR} * s_i(t) \quad (5\text{-}1)$$

針對 HRTF 摺積混和訊號，我們嘗試了前面章節所提出的三種分離方法，第一種是將聲譜圖當作影像訊號，利用 ISBS 作影像分離；第二種是用非線性投影（nonlinear projection, NP），將混和訊號投影到一個較為稀疏的領域，用 ISBS 解出白化矩陣和分離矩陣後再乘上原混和訊號；第三種則是利用非線性函數計算聲譜圖的每個 T-F unit 來自

於訊號源的貢獻程度，根據這些值對聲譜圖作非線性遮蔽（nonlinear masking, NM），把訊號源抽取出來。

在盲訊號分離法中，大多會用 SDR（Signal to Distortion Ratio）、SAR（Signals to Artifact Ratio）、SIR（Signal to Interference Ratio）作為效能的評比，經過演算法所分離出的訊號 $\hat{s}(t)$可分解成（式 5-2）：

$$\hat{s}(t) = s_{target}(t) + e_{interf}(t) + e_{noise}(t) + e_{artif}(t) \tag{5-2}$$

其中 $s_{target}(t)$為希望分離出的目標音源，$e_{interf}(t)$為目標音源以外的音源，也就是沒有分離乾淨所產生的干擾，$e_{noise}(t)$為音源以外的雜訊干擾，$e_{artif}(t)$為經由演算法所產生的人為干擾，如 musical noise。我們使用 C.F´evotte 所提供的 BSS_EVAL Toolbox[27]將 $\hat{s}(t)$分解後，SDR、SAR、SIR 的定義如下：

$$SDR = 10\log_{10}\frac{\|s_{target}\|^2}{\|e_{interf} + e_{noise} + e_{artif}\|^2} \tag{5-3}$$

$$SAR = 10\log_{10}\frac{\|s_{target} + e_{interf} + e_{noise}\|^2}{\|e_{artif}\|^2} \tag{5-4}$$

$$SIR = 10\log_{10}\frac{\|s_{target}\|^2}{\|e_{interf}\|^2} \tag{5-5}$$

因為我們的目標為語音分離，所以主要都會以 SIR 作為最重要的指標，SIR 值越高，表示來自其他音源的干擾很小，也就是分離的效果越好。

5.1 實驗設置

在各個模擬中，我們會分別針對音源來自不同角度的混和情況來做分離，音源來自右邊的時角度為 0°~ 90°，來自左邊時角度為 0°~ -90°（圖 5-2），以下我們用(θ_1,θ_2)來表示音源 1 來自 θ_1，音源 2 來自 θ_2。

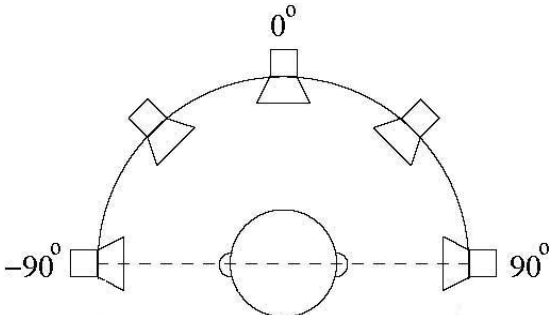


圖 5-2　訊號來源方向示意圖

第一個模擬實驗首先探討各個方法對於音源來自於頭的左右兩邊、以及來自於同一邊的結果，使用的語句為一男一女的中文語音，一組的角度為(45,-20)，另一組角度為(30,70)，下一節中會詳細的列出分離後的聲譜圖以及其 SAR、SDR、SIR 值，可以看出聲源方向對分離結果的影響。

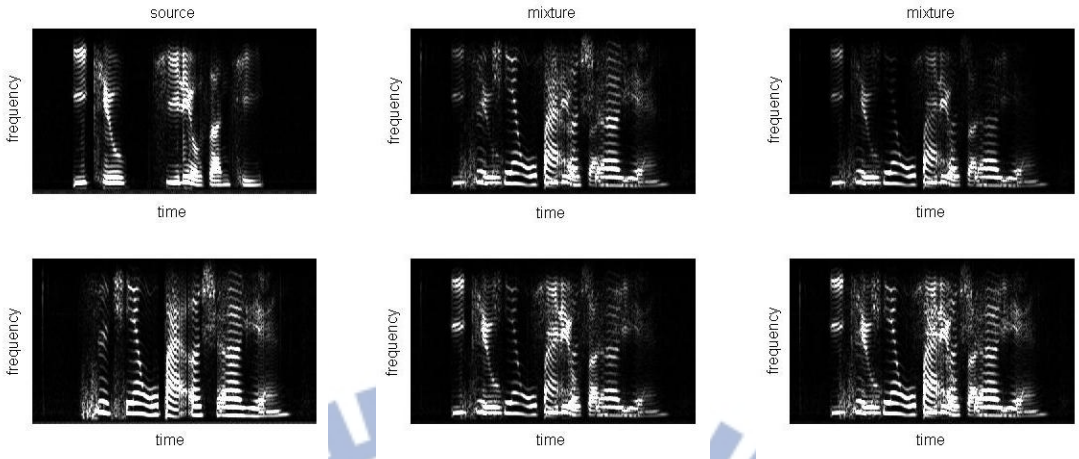
第二個模擬實驗為多個分離結果的平均值，我們使用 TIMIT 語料庫中的 12 句話，分別為 6 個男生與 6 個女生的語音，用其中兩個男生與兩個女生的語音當作音源 1，其他 4 個男生與 4 個女生的語音當作音源 2，如此總共會有 32 組混和訊號，最後用這 32 組混和訊號所解出的 SDR、SAR、SIR 平均值當作效能的評比。

第三個模擬實驗為欠定問題（unter-determined problem），也就是音源數量大於混和訊號數量，我們模擬 3 個音源來自不同的方位角，並比較 DUET 與 NM 的分離效果。

5.2 模擬結果

5.2.1 模擬一：一男一女分別位於(45,-20)、(30,70)

圖 5-3(a)為一男一女的音源聲譜圖，圖 5-3(b)為(45,-20)混和情況下的混和訊號，圖 5-3(c)為(30,70) 混和情況下的混和訊號；表 5-1 為兩種不同角度的混和情況下，混和訊號的 SIR。以下為三種方法之結果。



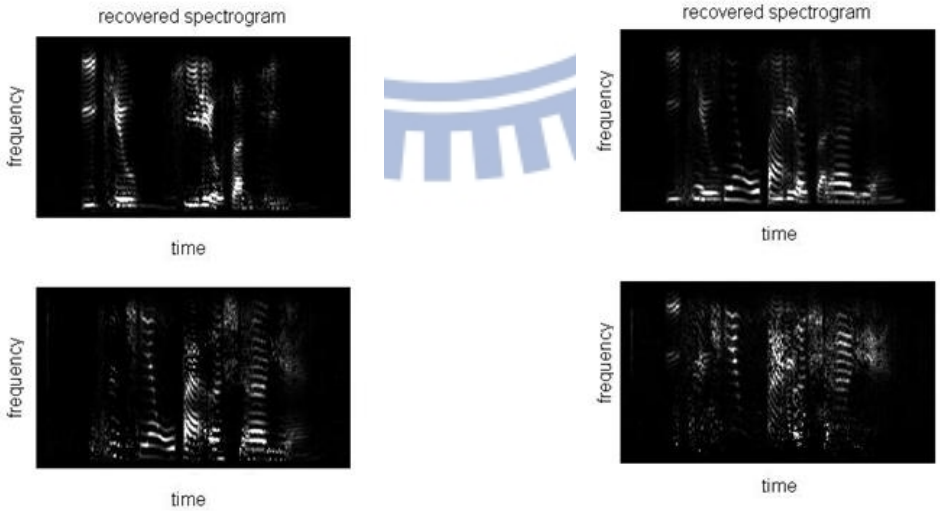
(a) (b) (c)

圖 5-3　音源與混和訊號之聲譜圖

表 5-1 混和訊號的 SIR

	(45,-20)		(30,70)	
mixture	x_1	x_2	x_1	x_2
SIR	-5.4592	-2.9114	-1.6474	3.3033

法一：ISBS 影像分離演算法



(a) (45,-20)分離出的聲譜圖　　(b) (30,70)分離出的聲譜圖

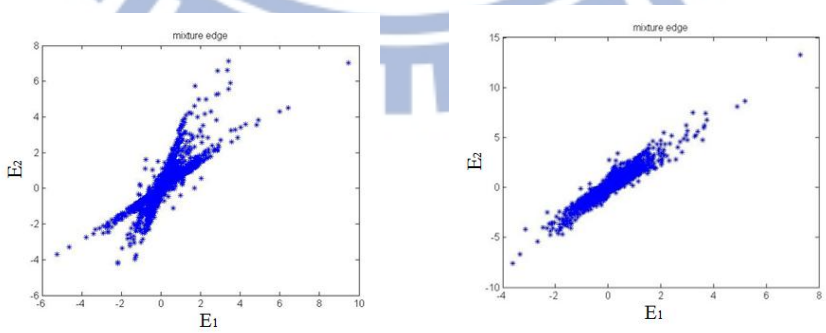
圖 5-4　ISBS 分離聲譜圖

表 5-2　ISBS 分離結果

	(45,-20)		(30,70)	
separated signal	y_1	y_2	y_1	y_2
SAR	10.1254	5.1701	-12.3089	13.8885
SDR	9.626	5.0818	-15.8396	1.725
SIR	19.6693	23.20041	-0.7369	2.1711

圖 5-4 為 ISBS 影像分離演算法對於兩種不同混和情況所分離出的聲譜圖，表 5-2 為其 SAR、SDR、SIR 值。可以看出當音源分為位於不同邊時可以有不錯的分離效果，但位於同邊時，分離效果不佳。

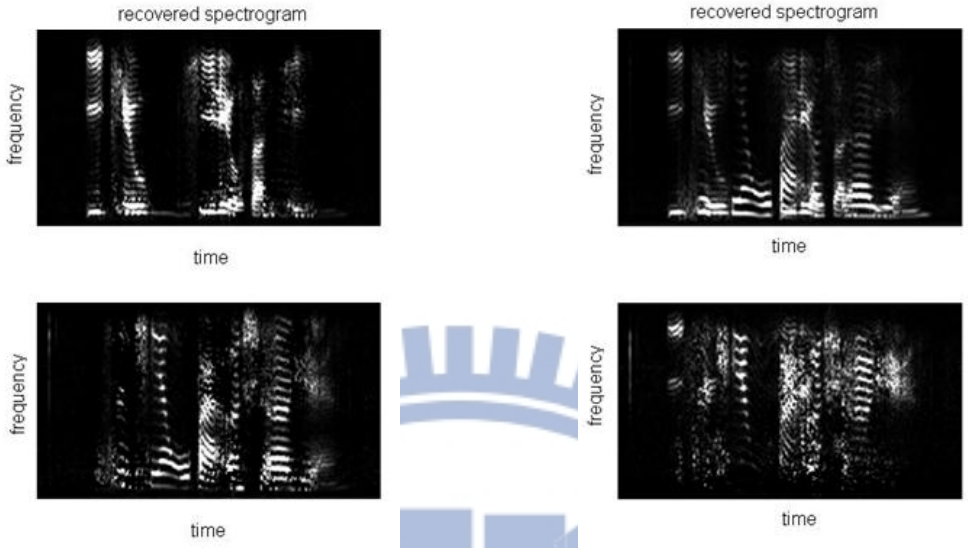
在(30,70)的混和情況下，兩個音源位在頭的同一邊，表示右耳所收到的混和訊號中兩個音源都比較大聲，左耳所收到的混和訊號中兩個音源都比較小聲，所以兩個混和訊號其實是很像的，我們以兩個很稀疏的語音混和後的情況為例，圖 5-5(a)為(45,-20)混和下 X_1 與 X_2 的邊緣圖(E_1, E_2)的散佈圖，圖 5-5(b)為(30,70)混和下 E_1 與 E_2 的散佈圖，可以看出在(45,-20)的情況下會有兩個不同的方向，但在(30,70)的情況下，兩個混和訊號的值趨於一致。在這種情況下，ISBS 的分離結果就沒有那麼好。



(a) (45,-20)之邊緣散佈圖　　(b) (30,70)之邊緣散佈圖

圖 5-5　混和訊號聲譜圖邊緣之散佈圖

法二：Nonlinear Projection



(a) (45,-20)分離出的聲譜圖　　(b) (30,70)分離出的聲譜圖

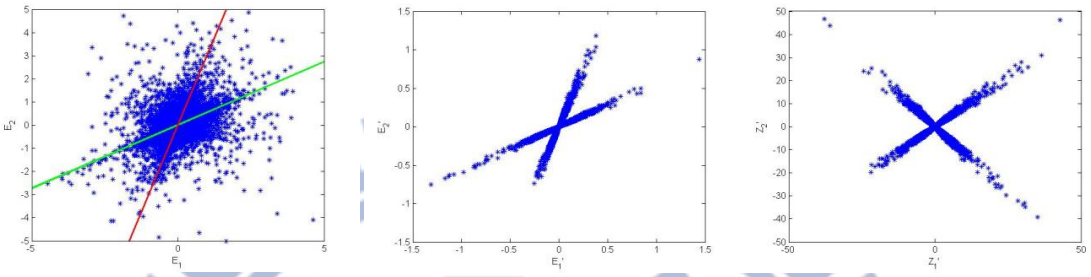
圖 5-6　NP 分離聲譜圖

表 5-3　NP 分離結果

	(45,-20)		(30,70)	
separated signal	y_1	y_2	y_1	y_2
SAR	-15.1541	-8.5498	-13.0611	13.9385
SDR	-15.4488	-8.5601	-16.3242	1.7358
SIR	11.6672	26.8111	-0.2822	2.1775

圖 5-6 為加入非線性投影後的影像分離演算法，在兩種不同混和情況下所分離出的聲譜圖，表 5-3 為其 SAR、SDR、SIR 值。可以看出此方法與只做影像分離的結果相差不大，都是在音源位於不同邊的情況下可分離，音源位於同邊時效果不佳。且雖然在(45,-20)狀況下的 SIR 值頗高，但因為訊號經過非線性處理，會使聲音變得比較不自然，因此 SAR 與 SDR 比原本的 ISBS 方法還要低。

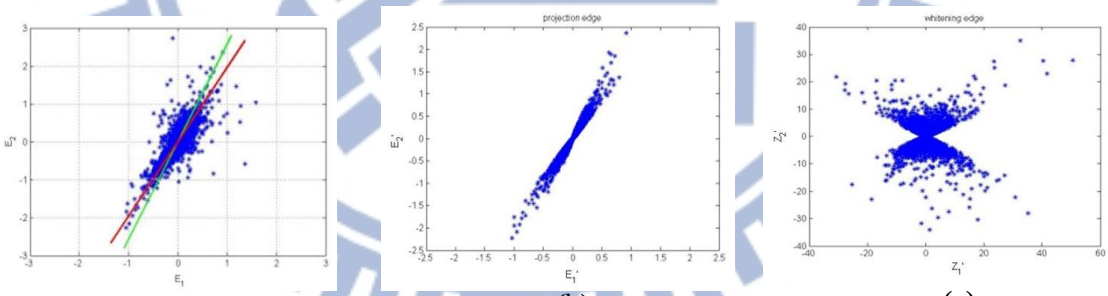
圖 5-7(a)為混和訊號聲譜圖的邊緣，E_1 與 E_2 的散佈圖，用 NPCM 找出兩個主要方向，圖 5-7(b)為使用非線性投影將所有訊號點投影到該方向上，圖 5-7(c)為圖 5-7(b)做白化處理後的結果，可以看出因為音源 1 和音源 2 位於頭的兩側，左右耳所收到的聲音有大小聲之分，所以方向會很明確，投影到兩個方向上的訊號點數量也很平均，因此在做白化處理後可以達到 de-correlation 的效果。



(a) (b) (c)

圖 5-7　音源位於不同邊之非線性投影效果

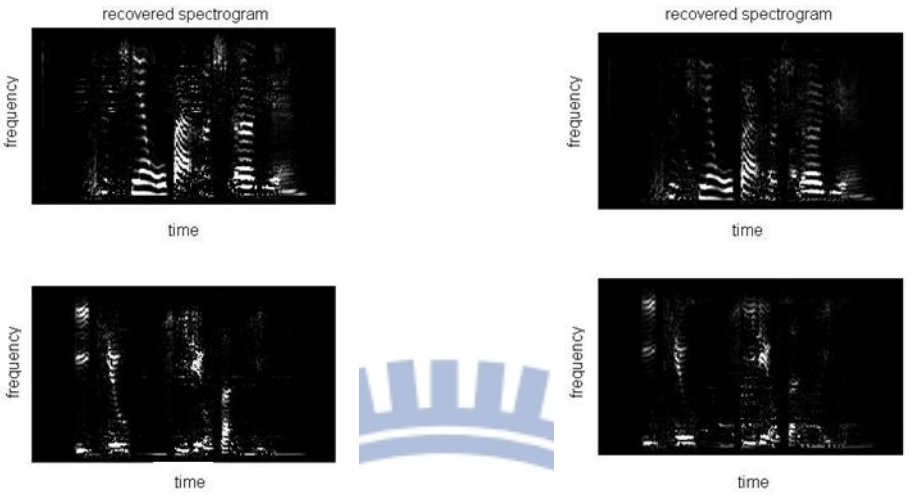
音源位於同邊的分離結果與 ISBS 類似，因為左右耳所收到的混和訊號很像，所以找出的兩個方向角度也很接近（圖 5-8(a)），做了非線性投影後，方向也很不明確（圖 5-8(b)），所以經過白化處理後，也很難達到 de-correlation 的效果（圖 5-8(c)）。



(a) (b) (c)

圖 5-8　音源位於同邊之非線性投影效果

法三：Nonlinear Masking



(a) (45,-20)分離出的聲譜圖　　(b) (30,70)分離出的聲譜圖

圖 5-9　NM 分離聲譜圖

表 5-4　NM 分離結果

	(45,-20)		(30,70)	
separated signal	y_1	y_2	y_1	y_2
SAR	6.8963	6.6124	6.2153	9.0356
SDR	6.8886	6.6009	5.6118	8.815
SIR	35.2297	33.2403	15.4121	22.3759

由圖 5-9 及表 5-4 的結果中可看出，此方法對於音源位於同邊的混和訊號也可以有很好的分離效果，在音源位於不同邊的混和情況，SIR 值也比前兩種方法要高。且此方法並沒有非線性的轉換，只是將混和訊號中來自音源的 T-F unit 抽取出來，因此對聲音的失真情形並不嚴重，SAR 和 SDR 皆為正值且每個分離出的訊號的情況皆一致。

5.2.2 模擬二：音源 1—2 男 2 女，音源 2—4 男 4 女，共 32 組混和訊號

以下圖表為三種方法在音源分別位於(-5,25)、(40,10)、(-70,-40)、(30,-30)、(15,50)混和下的分離結果，與 DUET 及 Convolutive ICA 所做的比較，數據為 32 組混和訊號的分離數據平均值。

表 5-5　(-5,25)分離結果

(-5,25)	s_1			s_2		
	SAR	SDR	SIR	SAR	SDR	SIR
DUET	10.58	10.40	25.33	-11.84	-13.49	9.09
cICA	6.82	4.96	10.71	3.75	3.67	30.46
ISBS	2.81	1.76	16.46	5.53	2.69	13.10
NP	7.48	3.47	13.96	0.08	-8.06	4.93
NM	8.41	8.26	24.84	8.56	8.39	24.16

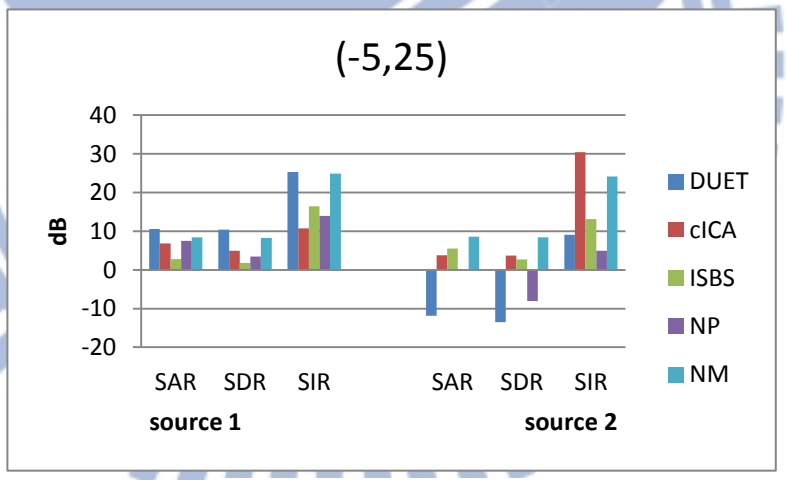


圖 5-10　(-5,25)分離結果

表 5-6　(40,10)分離結果

(40,10)	s_1			s_2		
	SAR	SDR	SIR	SAR	SDR	SIR
DUET	-10.88	-11.33	13.07	8.79	8.55	24.27
cICA	0.97	-1.28	10.50	5.93	4.92	13.16
ISBS	2.81	-5.68	10.99	4.62	0.49	7.90
NP	-0.48	-5.10	12.30	5.51	-0.93	9.95
NM	9.39	9.22	24.44	8.10	7.91	23.67

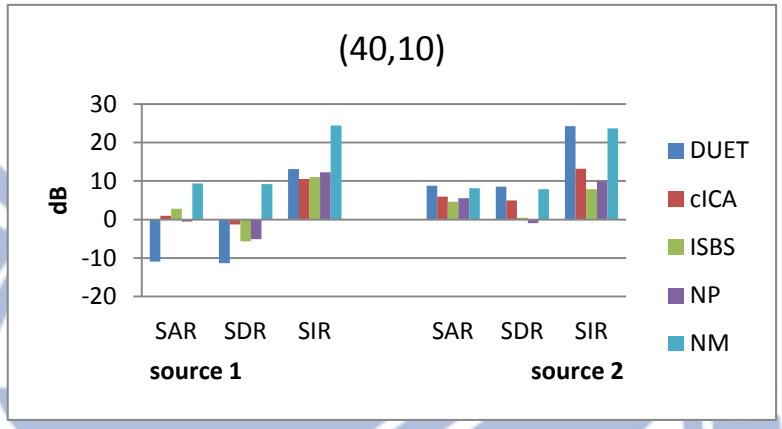


圖 5-10　(40,10)分離結果

表 5-7　(-70,-40)分離結果

(-70,-40)	s_1			s_2		
	SAR	SDR	SIR	SAR	SDR	SIR
DUET	6.45	5.79	21.87	-11.46	-13.24	10.80
cICA	-2.63	-4.68	9.43	-1.91	-4.13	6.25
ISBS	-2.65	-5.96	2.80	-0.89	-7.25	1.97
NP	-5.09	-8.58	2.90	4.79	-9.20	-2.34
NM	-3.59	-4.97	10.04	-7.06	-12.71	-0.41

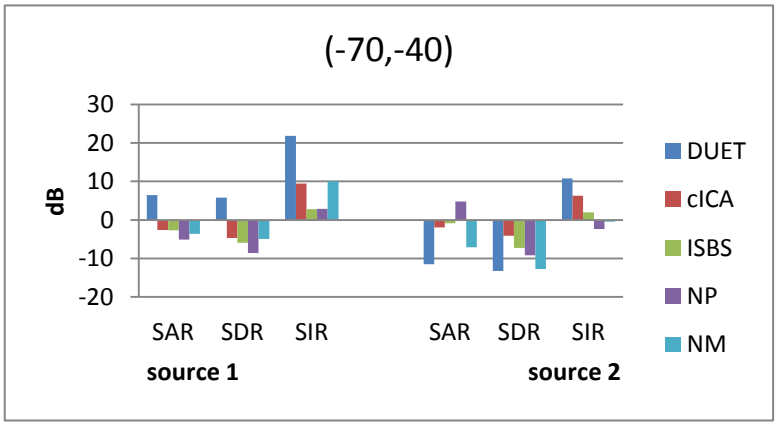


圖 5-10　(-70,-40)分離結果

表 5-8　(-30,30)分離結果

(-30,30)	s_1			s_2		
	SAR	SDR	SIR	SAR	SDR	SIR
DUET	-20.13	-20.39	17.07	9.46	8.48	23.96
cICA	8.04	7.31	17.37	7.68	7.09	18.79
ISBS	-1.51	-2.42	21.80	-4.09	-9.98	8.29
NP	-2.59	-7.57	10.49	-0.16	-5.69	13.57
NM	9.17	8.82	25.79	8.95	9.05	26.54

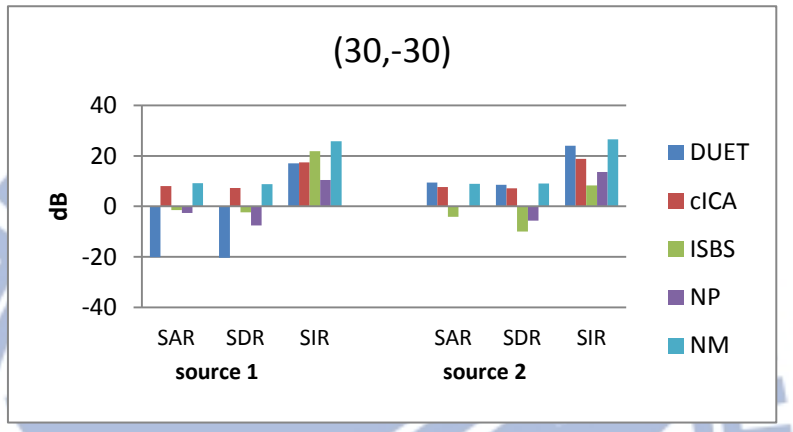


圖 5-11　(30,-30)分離結果

表 5-9　(15,50)分離結果

(15,50)	s_1			s_2		
	SAR	SDR	SIR	SAR	SDR	SIR
DUET	7.00	3.09	18.11	-6.59	-7.48	21.41
cICA	4.64	3.74	13.47	-2.05	-2.44	16.17
ISBS	1.83	-2.46	4.49	5.18	-4.92	15.70
NP	1.11	-3.87	3.99	-2.46	-5.20	17.57
NM	6.09	5.11	22.50	5.27	5.90	21.76

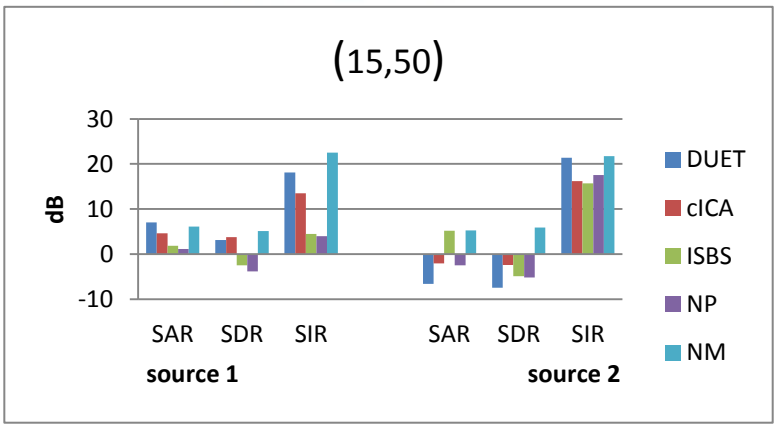


圖 5-12　(15,50)分離結果

（表 5-10）為只看 SIR 的結果，其值為 s_1 的 SIR 與 s_2 的 SIR 平均。

表 5-10　五種方法的 SIR 平均值

SIR	(-5,25)	(40,10)	(-70,-40)	(30,-30)	(15,50)
DUET	17.21	18.68	16.33	20.52	19.76
cICA	20.58	11.83	7.84	18.08	14.82
ISBS	14.78	9.45	2.38	15.05	10.09
NP	9.45	11.13	0.28	12.03	10.78
NM	24.50	24.06	4.82	26.16	22.13

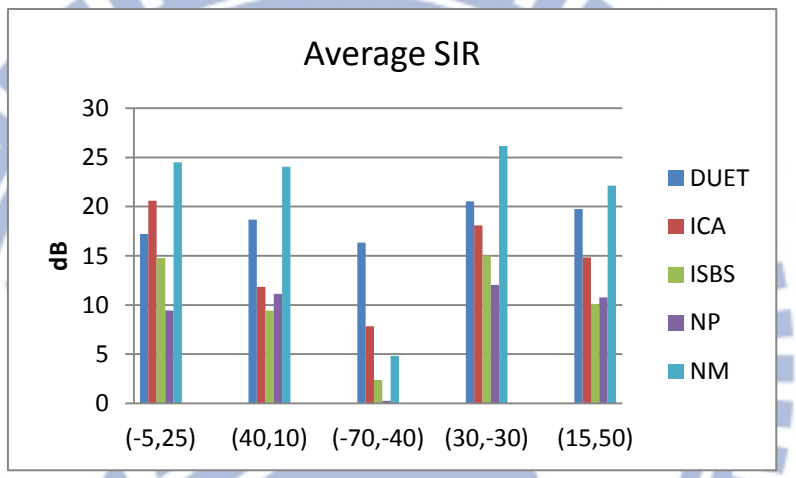


圖 5-13　五種方法的 SIR 平均值

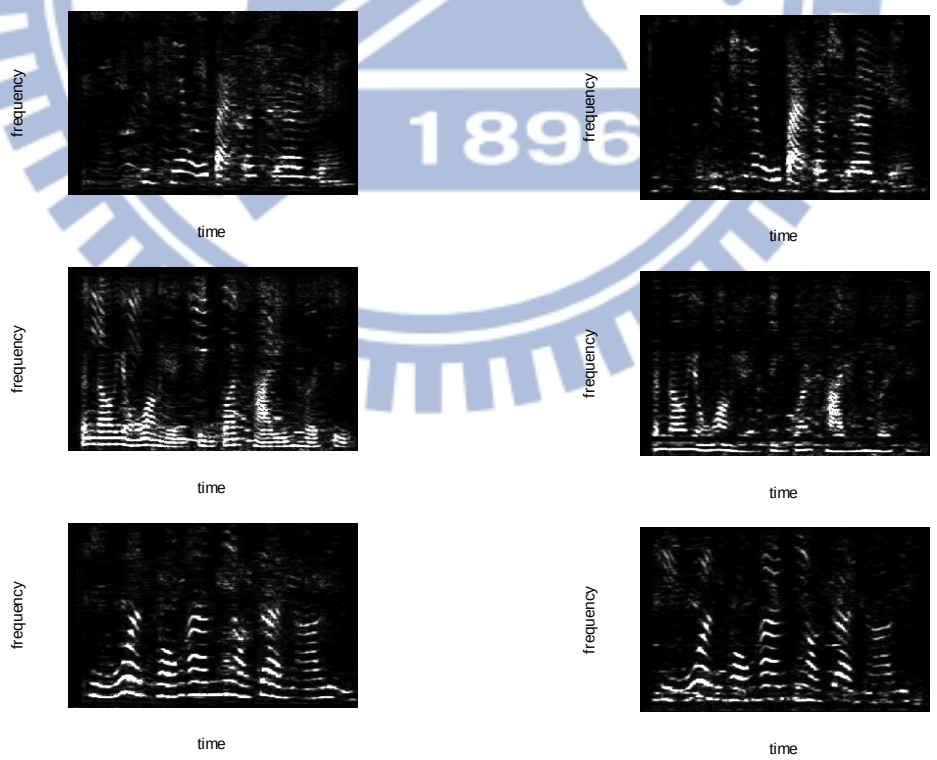
與模擬一的結果類似，ISBS 及 NP 演算法在音源位於不同兩邊時分離結果較好，在 ISBS 本身可分離的情況下，加入 NP 處理後效果稍差；但當音源位於同邊時，ISBS 的分離結果較差，加入非線性投影後結果會稍好。Convolutive ICA 同樣也是在音源位於同邊時結果較差，而 DUET 則是在各種角度混和下皆有穩定的分離效果。NM 在大部分的情況下皆有最高的 SIR 值，且 SAR 與 SDR 值皆很穩定，惟有在(-70,-40)的混和情況下，NM 只能分離出其中一個音源，另一音源 SIR 值很低，可能原因為 NM 考量的是兩個混和訊號中音源大小聲的差異，當某一音源來自-70°，表示在右耳的混和訊號 $x_R(t)$ 中，此音源的音量是很小的，如此一來在散佈圖中會比較難辨認方向，且使用遮蔽的方法也比較難將此音源抽取出來。而 DUET 演算法中不只考量音量大小，同時考慮每個 T-F unit 之相位差，因此在(-70,-40)的混和情況下，依然有穩定的分離效果。

5.2.3 模擬三：3 個音源分別位於 (40,-25,10)

在盲訊號分離法中，當音源數量大於混和訊號數量時，因無法使用反矩陣的解法，大多也是使用頻譜圖遮蔽的方式,因此五種方法中只有 DUET 與 NM 演算法可以解決欠定問題,以下為兩種方法的分離結果。由（圖 5-14）中可看出兩種方法皆可分離出音源，（表 5-12）中可以看出 NM 對於其中兩個聲源 s_1 與 s_2 有較好的分離效果，但在 3 個音源的情況下，遮蔽矩陣的計算比較容易受到其他訊號點的影響，因此 s_3 的分離結果相對 DUET 稍差。

表 5-12　DUET 與 NM 的分離結果

	DUET			NM		
	s_1	s_2	s_3	s_1	s_2	s_3
SAR	-13.88	2.74	8.45	-11.75	-1.39	4.64
SDR	-14.95	19.68	20.94	-12.01	-1.46	4.27
SIR	5.73	19.68	20.94	12.58	20.05	16.45



(a) DUET 分離結果　　(b) NM 分離結果

圖 5-14　欠定問題所分離出的聲譜圖

5.3 結論

本論文探討盲訊號分離法對於語音分離的成效，主要針對以 HRTF 為脈衝響應的摺積混和訊號來做分離，以模擬音源來自不同方位時，人的左右耳所收到的混和訊號。考慮摺積混和訊號無法直接以時域訊號來做分離，論文中實驗了三種不同的方法，將訊號轉到頻域後，再以不同特性的演算法加以考量。第一種方法為 ISBS 影像分離演算法，也就是將混和訊號的聲譜圖作為影像訊號來處理，利用影像的邊緣圖具有稀疏的特性，以最大化影像邊緣圖作為演算法的目標，並以 FastICA 作為疊代法的初始條件計算，將解出的分離矩陣乘上原聲譜圖，就可以得到分離出的音源聲譜圖，最後再重建回聲音。第二種方法在 ISBS 中加入非線性投影的前置處理，利用非線性投影將影像的邊緣訊號轉成較為稀疏的訊號，希望其所解出的白化矩陣與分離矩陣能有比較好的分離效果。第三種方法將聲譜圖分頻，利用每個頻帶的混和訊號皆很稀疏的特性，使用非線性函數計算音源對於混和訊號的貢獻值，遮蔽掉貢獻值較小的訊號點，用非線性遮蔽的方法將音源抽取出來。

比較三種方法的結果，在音源位於不同邊的混和情況下，三種方法皆可以有不錯的分離效果，其中又以非線性遮蔽的方法 SIR 值最高，且 SAR 與 SDR 值也很穩定。在音源位於同邊的混和情況下，因為左右耳所收到的混和訊號趨於一致，ISBS 演算法無法分離，即使加入非線性投影，也因為無法成功達到 de-correlation 的效果，分離效果依然不佳；而非線性遮蔽則是只要左右耳的訊號有大小聲的差異，便可以利用此差異將訊號分離出來，因此對於同邊的混和訊號依然可以做分離。與現有的 Convolutive ICA 及 DUET 演算法來做比較，非線性遮蔽在不同的混和角度下，分離效果穩定且良好，惟有在某一音源太偏向正左方或正右方時，經過 HRTF 模擬後另一耳的音量太小，造成演算法沒辦法分出此音源，也就是只能分離出其中一個音源。

參考文獻

[1] A. Hyvärinen, J. Karhunen, E. Oja, *Independent Component Analysis*, John Wiley, 2001.

[2] James V. Stone, *Independent Component Analysis: a Tutorial Introduction*, MIT Press, 2004.

[3] P. Comon, "Independent Component Analysis, a new concept?" Signal Processing, Elsevier, 36(3):287–314, April 1994.

[4] A. Hyvärinen and E.Oja, "A fast fixed-point algorithm for independent component analysis. Neural Computation, 9(7):1483-1492, 1997.

[5] A. Hyvärinen, "Fast and robust fixed-point algorithms for independent component analysis," IEEE Trans. on Neural Networks, 10(3):626-634,1999.

[6] A. Hyvärinen, "Survey on independent component analysis," Neural Computing Surveys, 2:94-128, 1999.

[7] E. Bingham and A. Hyvärinen, "A fast fixed-point algorithm for independent component analysis of complex-valued signals." Int. Journal of Neural Systems, 10(1):1–8, 2000.

[8] A. J. Bell and T. J. Sejnowski, "A non-linear information maximization algorithm that performs blind separation," In Advances in Neural Information Processing Systems 7, pages 467–474. The MIT Press, Cambridge, MA, 1995.

[9] A. J. Bell and T. J. Sejnowski, "An information-maximization approach to blind separation and blind deconvolution," Neural Computation, 7:1129-1159, 1995.

[10] M. Gaeta and J. L. Lacoume, "Source separation without prior knowledge: the maximum likelihood solution," In Proc. EUSIPCO'90, p. 621–624, 1990.

[11] B. A. Pearlmutter and L. C. Parra, "Maximum likelihood blind source separation: A context-sensitive generalization of ICA," In Advances in Neural Information Processing Systems, volume 9, p. 613–619, 1997.

[12] T. W. Lee, M. Girolami, and T. J. Sejnowski, “Independent component analysis using an extended infomax algorithm for mixed sub-gaussian and super-gaussian sources,” Neural Computation, 11(2):417–441, 1999.

[13] P. Smaragdis, “Blind separation of convolved mixtures in the frequency domain.” Neuro computing, 22:21–34, November 1998.

[14] S. Ikeda and N. Murata, “A method of ICA in time-frequency domain,” in Proceedings of International Workshop on Independent Component Analysis and Blind Signal Separation (ICA '99), p. 365–371, Aussions, France, January 1999.

[15] N. Murata, S. Ikeda, and A. Ziehe, “An approach to blind source separation based on temporal structure of speech signals,” Neurocomputing, vol. 41, p.1-24, 2001.

[16] H. Sawada, R. Mukai, S. Araki, and S. Makino, “A robust and precise method for solving the permutation problem of frequency-domain blind source separation,” Proc. Int. Symp. Independent Component Analysis Blind Signal Separation (ICA2003) , p.505-510, 2003.

[17] P.D.O’Grady, B.A.Pearlmutter, and S.T.Rickard, “Survey of sparse and non-sparse methods in source separation,” International Journal of Imaging Systems and Technology (IJIST), vol.15, p.18–33, 2005.

[18] P. Bofill and M. Zibulevsky, “Underdetermined blind source separation using sparse representations ,” Signal Processing, vol. 81, no. 11, p.2353-2362, 2001.

[19] M. Zibulevsky, P. Kisilev, Y. Y. Zeevi, and B. A. Pearlmutter, “Blind source separation via multinode sparse representation,” In Advances in Neural Information Processing Systems 14, p.1049-1056, MIT Press, 2002

[20] A. Jourjine, S. Rickard, and O. Yilmaz, “Blind separation of disjoint orthogonal signals: Demixing N sources from 2 mixtures,” In IEEE Conference on Acoustics, Speech, and Signal Processing (ICASSP2000), vol.5, p.2985-2988, 2000.

[21] D. D. Lee and H. S. Seung, “Learning the parts of objects by non-negative matrix factorization.” Nature, 401(675), 788-791, 1999.

[22] P. O. Hoyer, “Non-negative matrix factorization with sparseness constraints,” The

Journal of Machine Learning Research, 5, p.1457-1469, 2004.

[23] K. Kayabol , E. E. Kuruoglu and B. Sankur, “Bayesian separation of images modeled with MRFs using MCMC,” IEEE Trans. Image Processing, vol. 18, p.982 , 2009.

[24] W.Souidene, A. Aissa-El-Bey, K. Abed-Meraim, A. Beghdadi, “Blind image separation using sparse representation,” IEEE International Conference on Image Processing. Vol. 3, pp.III-125–III-128, 2007.

[25] S. Araki, R. Mukai, S. Makino, “The fundamental limitation of frequency-domain blind source separation for convolutive mixtures of speech,” IEEE Trans. Speech Audio Process., vol.11, p.109-116, 2003.

[26] Z. Guoxu, Y. Zuyuan, X. Shengli, Y. Jun-Mei, “Mixing matrix estimation from sparse mixtures with unknown number of sources,” IEEE Transactions on Neural Networks, vol.22, p.211-221, 2011.

[27] C. F´evotte, R. Gribonval and E. Vincent, BSS EVAL Toolbox User Guide, IRISA Technical Report 1706, Rennes, France, April 2005. http://www.irisa.fr/metiss/bsseval/.