國 立 交 通 大 學
電機與控制工程研究所

碩 士 論 文

以雙核心平台實現
即時影音追蹤與語音純化系統

Implement a real-time Human Face/Sound Source Tracking and Speech Purification System on a Dual-Core platform

研 究 生： 黃 啟 揚
指導教授： 胡 竹 生 教授

中 華 民 國 九 十 七 年 七 月


E
S
A
1896

以雙核心平台實現
即時影音追蹤與語音純化系統

Implement a real-time Human Face/Sound Source Tracking and Speech Purification System on a Dual-Core platform

研 究 生：黃 啟 揚　　Student：Chi-Young, Huang
指導教授：胡 竹 生 教授　　Advisor：Prof. Jwu-Sheng, Hu

國立交通大學
電機與控制工程學系
碩 士 論 文

A Thesis
Submitted to Institute of Electrical and Control Engineering
College of Electrical Engineering and Computer Science
National Chiao Tung University
in partial Fulfillment of the Requirements
for the Degree of Master
in

Electrical and Control Engineering

July 2008

Hsinchu, Taiwan, Republic of China

中 華 民 國 九 十 七 年 七 月


E
S
A
1896

以雙核心平台實現
即時影音追蹤與語音純化系統

研究生：黃　啟　揚　　指導教授：胡　竹　生　教授

國立交通大學電機與控制工程研究所碩士班

摘　要

本論文提出一套以嵌入式雙核心平台實現之人臉追蹤、聲源方位估測與語音純化系統。人臉追蹤系統可針對人臉特徵做持續且即時性的追蹤，聲源方位估測系統則可找出發聲者所在方位，而語音純化系統可強化使用者方位語音、抑制其他方位噪音，優化語音品質。本系統在硬體上選用 TI 推出的嵌入式雙核心系統 DM6446 EVM 為發展平台，平台 DSP 核心負責演算法運算，而 ARM 核心主要負責系統周邊控制。影像資訊透過 PTZ 攝影機擷取，而聲音擷取則使用實驗室開發的數位式麥克風陣列訊號擷取系統以擷取多通道聲音資訊。軟體上整合了影、音相關演算法，包括語音活動偵測演算法(VAD, Voice Activity Detection)、聲源方位估測演算法(MUSIC, Multiple Signals Classification Method)、適應性語音純化演算法(Adaptive Beamformer)與平均位移演算法(Mean-Shift)，希望藉此建置兼具聽覺與視覺之人機互動介面，具有視訊會議系統、居家保全系統和機器人..等相關應用面。

Implement a real-time Human Face/Sound Source Tracking and Speech Purification System on a Dual-Core platform

Student： Chi-Young, Huang Advisor： Prof. Jwu-Sheng, Hu

Institute of Electrical and Control Engineering

ABSTRACT

The thesis describes an implementation of human face tracking, sound source direction estimation and speech purification on a dual-core platform. The system can perform real-time tracking of human face, estimating the sound source direction and enhance the speech in that direction while depress the noise in other directions. The development platform is TI DM6446 EVM which is an embedded dual-core system. DSP core is responsible for algorithm realization. ARM core is responsible to control the system peripherals. The image is captured by a PTZ camera and the sound data is acquired by digital microphone array signal acquisition system to get multi-channels sound data. The system software integrates Voice Activity Detection Algorithm (VAD), Multiple Signals Classification Method (MUSIC), Adaptive Beamformer and Mean-Shift Object Tracking Algorithm. Using the technique, we can build a human-robot interface with vision and hearing. This system can apply to video conference, home guarding and robot etc.

誌 謝

噢耶~ 論文終於完成了，真是非常的開心，在開心之餘，首先要感謝我的指導教授胡竹生老師，每當我研究上遇到問題而停滯不前時，老師總能像指南針一樣指引我正確的研究方向，讓我能夠繼續向前，非常謝謝老師的指導，另外，每次實驗室開會時，都可以看到老師展現出充滿創意的研究想法、深厚的專業知識與科學的分析問題的能力，令我敬佩不已，不禁讚歎起老師你還真是強阿，我真是跟對老師了~ 歐耶。

興哥(佳興學長)不論是在計畫執行上、研究學習上、TI 比賽上，總給予我很大的幫助，非常的感謝。是法哥(崇維學長)帶領我接觸嵌入式系統的，很感謝您。古語有云: ”軟體問法哥、硬體問永融”，所以帶領我勇闖東元比賽的永融學長也是一定要感謝的。立偉學長帶我大學專題，幫助我推甄上研究所進入 XLAB，這也是一定要感謝的。寫到這裡，我深深覺得要感謝的人真的實在太多了，因為這個實驗室一起研究學習的夥伴們實在太酷了: 電影健身人劉大學長、幽默風趣人宗敏學長、美麗冰箱人鏗元學姊(冰過真的比較好吃嗎?)、聯誼大師兄勁源學長(感謝您辦的聯誼歐^_<)、實驗室魔獸三國大隊長 PLE(你是伐木工嗎?)、實驗室魔獸三國副隊長 dowind (哪有劉備一直追著別人砍的拉 XD)、努力認真研究且常幫我 DEBUG 的發狂排球強迫症人 PAPA、團康唱歌麻將樣樣行之無敵包租公 GUM、忠厚老實愛老婆人俊宇、樣樣第一書卷大胃王 95 吉、唱歌 A 咖摔倒人瓊文(哈哈哈)、藤源拓海飆車人唐哥、英文很強深綠人嘟嘟(你是浩克嗎?)、把妹時尚穿衣人 LUNDY、第一神拳打暈人 JUDO、打球靈活認真人肉鬆、弄錢一流助理人淑伶，大家都好酷喔，讓我在實驗室的日子多采多姿，真的很感謝實驗室的大家。

我要感謝我的家人，老爸黃大岩與老媽張穗櫻在我的背後一直支持著我，若是沒有你們我是不可能完成學業的，最後還有感謝在天國的阿嬤，你超疼我的，真希望你在天上能夠看到我畢業了。

目錄

表列

圖列

第一章　緒論

1.1 研究動機

隨著嵌入式系統應用的處理器在速度及功能上日益進步，嵌入式系統可說是未來生活的基礎平台，此項技術的蓬勃發展與各式各樣的創新思維，帶來了多媒體應用面的發展有成與生活方式的革新。在眾多的多媒體資訊中，多模式介面的人機互動將在未來生活方式中佔據極重要的地位，有鑑於此本論文將發展並提出一套具有雙模式聽覺與視覺的人機互動系統。

人臉追蹤往往會伴隨著一個問題，便是影像脫鎖與對未進入 CCD 鏡頭前對影像位置之估測。過去在影像未曾進入 CCD 影像擷取區間時，往往無法對影像位置作估測，因而可能是以一定路徑規則去搜尋影像。但此方式較為費時，因此希望可以透過聽覺系統的幫助輔助視覺，本論文使用麥克風陣列訊號處理技術來解決此問題，系統利用聲音處理子系統判別真人語音並對發聲源位置作估測，將此位置資訊提供給攝影機伺服器作為從動角度之依據，進而可快速鎖定人聲位置。透過對本系統的呼叫，攝影機可快速鎖定人臉，省去以一定路徑規則搜尋影像的步驟，達到聽覺系統輔助視覺系統的功能。

環境中的語音訊號干擾源總是存在，例如冷氣機、喇叭、電腦風扇、密閉空間反射等等。當語音訊號遭受干擾時，若用於語音辨識中、辨識率會大為降低，若用於通訊中，通話品質也大受影響。因此本論文希望利用麥克風陣列技術來對語者語音做純化的動作，以降低干擾源對語音訊號的影響，並將提高訊噪比(SNR)後的語音訊號做即時性的語音輸出。

總結以上，本論文將整合軟硬體技術，建構一套語音活動偵測、聲源方位估測、語者語音純化與人臉追蹤之系統。未來此系統不僅可用於微軟 Vista 作業系統中麥克風陣列的應用、視訊會議系統上之 Walk and Talk 功能，或應用於機器人產業人機互動，作為一個聲影辨識合作的最佳範例。

1.2 研究目標

本論文研究目標如下:

1. 整合數位式麥克風陣列訊號擷取系統與嵌入式雙核心系統平台 DM6446 EVM，解決跨平台間資料傳輸問題。
2. 在 DM6446 的 DSP 核心上實現語音活動偵測演算法，使系統具有判斷何時有使用者說話的功能。
3. 在 DM6446 的 DSP 核心上實現聲源方位估測演算法，使系統具有對發聲源的水平方位作即時性的估測的功能。
4. 在 DM6446 的 DSP 核心上實現適應性語音純化演算法，使系統可抑制背景環境噪音，強化語者語音資訊，達到提高訊噪比(SNR)的目的。
5. 以 PTZ 攝影機擷取影像，在 DM6446 的 ARM 核心上實現平均位移演算法，使系統具有人臉追蹤的功能。
6. 整合起影像與聲音系統軟硬體,在平台上建立一套雙模式聽覺與視覺的人機互動系統。

1.3 論文架構

整篇論文大致上可以切分為兩個部份，分別是第二章的系統原理分析與第三章的系統軟硬體設計與實現。系統原理分析這章節會對系統所使用到的相關影、音演算法的原理做介紹，系統軟硬體設計與實現這章節則會針對實驗平台的軟硬體開發環境、系統的軟硬體架構與多通道聲音傳輸架構作一個說明。最後的第四章節會呈現系統各項功能的測試結果並對研究成果作個結論。

第二章　系統原理分析

2.1 語音活動偵測系統

語音活動偵測 VAD (Voice Activity Detection)是用來判定是否有真人語音，近年來已廣泛用於通訊上達到節省能量耗損的目的。若用於語音辨識方面是屬於語音辨識的前處理，對辨識結果的影響很大，精確的語音活動偵測可降低噪音影響並提高辨識率。傳統的語音活動偵測大多使用語音能量或過零率（zero-crossing rate）等資訊來判別，本節將介紹的語音活動偵測演算法是使用長時間語音資訊（long-term speech information）來判別是否有真人語音[1]。

最常見的判定真人語音資訊為語音能量和過零率，雜訊及氣音的過零率都很高，語音能量都較低。例如，由歐洲電信標準協會（ETSI）所制定用於 GSM（Global System for Mobile Communications）系統中的 AMR（Adaptive Multi Rate）VAD 判定方法就採用了能量、週期、頻譜失真等三種參數來判定[2-3]。另外由國際電信聯盟（ITU）所制定的 G.729-VAD 採用了全頻帶能量差、低頻帶能量差、頻譜失真和過零率四種參數來判定[4-5]。論文中使用的 VAD 演算法是使用長時間語音的資訊而非傳統瞬間音框（instantaneous frame）資訊，針對長時間語音資訊，定義出下列定義：

- **Long-Term Spectrum Envelope (LTSE)**

  若 x（n）為一段包含有雜訊的語音訊號，而 X（k,l）代表著 x（n）中第 l 個音框第 k 個頻率的值，那麼 N 階的 LTSE 定義為：

  $$\mathrm{LTSE}_{\mathrm{N}}(k,l)=\max\{X(k,l+j)\}_{j=-N}^{j=+N} \quad (1)$$

  其 $\mathrm{LTSE}_{\mathrm{N}}(k,l)$ 代表的意義為，從第 l-N 個音框(frame)到第 l+N 個音框，這 2N+1 個音框分別對其取頻譜絕對值（Amplitude Spectrum）後，在第 k 個頻率下，取這 2N+1 個頻域絕對值內的最大值。而 LTSE 則代表了長時間語音資訊的意義，因為 LTSE 不只是對單一

音框取值，而是針對 2N+1 個音框取最大值，這樣的好處是不容易忽略某些字頭的子音或是摩擦音。除了 LTSE 外，為了判定是否為真人語音，必須定義另一項定義 LTSD。

- **Long-Term Spectral Divergence (LTSD)**

LTSD 的定義如（2）式：

$$\mathrm{LTSD}_{\mathrm{N}}(l)=10\log_{10}\left(\frac{1}{NFFT}\sum_{k=0}^{NFFT-1}\frac{LTSE^{2}(k,l)}{N^{2}(k)}\right) \tag{2}$$

其中 NFFT 代表了作 FFT(Fast Fourier Transform)的點數，而 $N\ (k)$ 代表了雜訊的頻譜絕對值平均，定義如(3)式：

$$N_{K}(k)=\frac{1}{2K+1}\sum_{j=-K}^{j=K}X(k,l+j) \tag{3}$$

從（3）式可看出，$N_{K}(k)$代表在第 k 個頻率下，第 l 個音框及前後 K 個音框的頻譜絕對值平均，$X(k,\mathrm{l})$和先前定義一樣，代表現階段語音的頻譜絕對值。因此 LTSD 的意義為：現階段長時間語音的頻譜能量佔了雜訊頻譜能量的比例，換句話說判定是否為真人語音是用了現階語音能量的大小來判定，而此能量大小包含了長時間語音資訊，並非只有單一音框資訊。當 LTSD 大於某個臨界值則判定為真人語音，反之則非真人語音，而此臨界值γ定義如下：

$$\gamma=\begin{cases}\gamma_{0} & E\le E_{0}\\ \gamma_{0}+\dfrac{\gamma_{1}-\gamma_{0}}{E_{1}-E_{0}}(E-E_{0}) & E_{0}<E<E_{1}\\ \gamma_{1} & E\ge E_{1}\end{cases} \tag{4}$$

其中E_{0}和E_{1}代表了在最乾淨和最吵雜的情況下，雜訊的能量，而 E 是指現階段雜訊的能量。γ_{0}和γ_{1}代表在最乾淨和最吵雜的情況下與 LTSD 比較的臨界值，因此E_{0},E_{1},γ_{0}和γ_{1}是先設定好的初始值。從（4）式可觀察出當現階段雜訊能量介於E_{0}和E_{1}時，則γ會依$E-E_{0}$在$E_{1}-E_{0}$所佔的比例，作出γ_{0}的線性調整。

■ **VAD 系統流程**

而 VAD 演算法的流程如圖 2-1 所示：

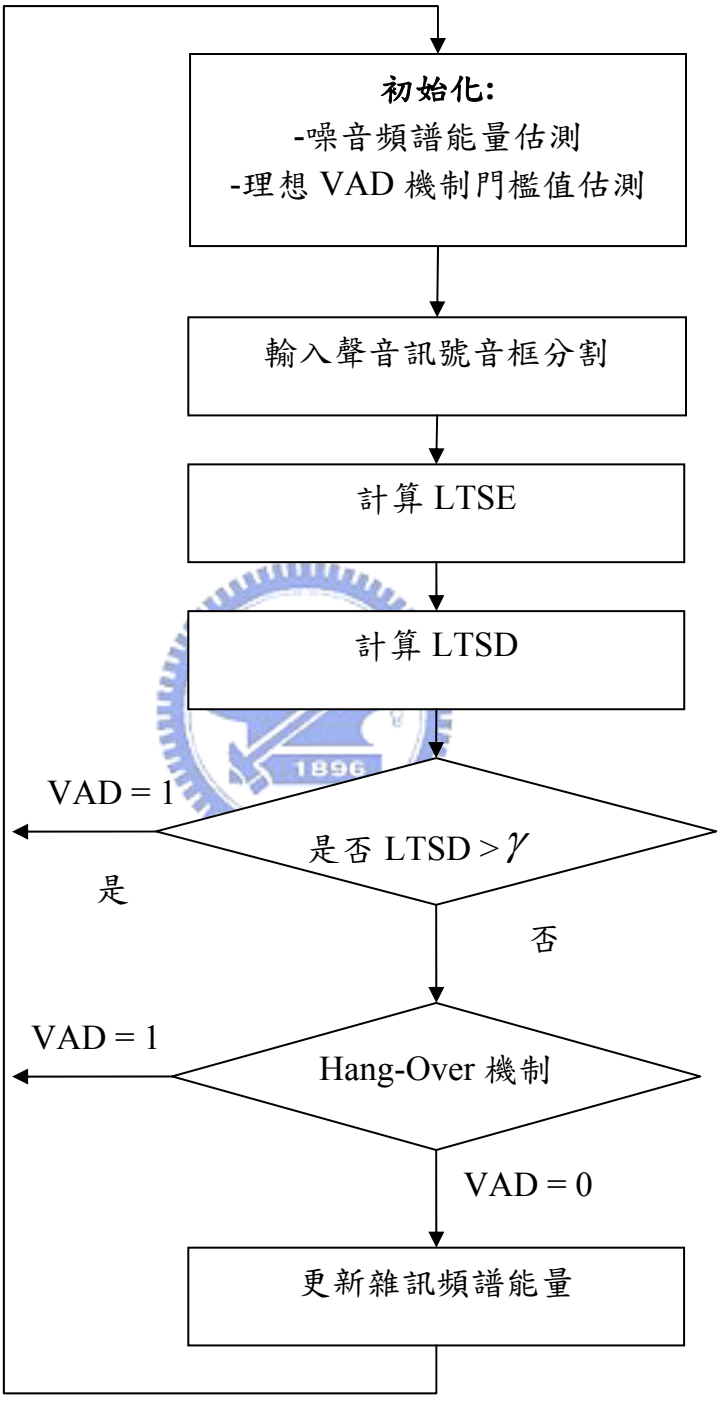


圖 2-1　VAD 演算法流程圖

1. 設定初始值E_0, E_1, γ_0和γ_1。
2. 將語音作切割，一個音框(frame)為 30ms，而音框和音框的交疊為 20ms。
3. 計算 LTSE 和 LTSD。
4. 將 LTSD 與γ作比較，若 LTSD > γ則判定為真人語音，若 LTSD < γ，則經過 Hang-Over 機制。
5. 經過 Hang-Over 機制，若為非真人語音，則更新雜訊頻譜絕對值平均$N(k)$。

Hang-Over 機制是為了延長字母尾音判定為真人語音的機制，因為字母尾音部分通常能量較小，容易被判定為非真人尾音，因此系統中加入 Hang-Over 機制，彌補字母尾音能量小的問題。另外在更新雜訊頻譜絕對值平均$N(k)$方面，並非完全的更新，而是利用了適應性訊號處理的觀念，定義如下：

$$N(k,l) = \alpha N(k,l-1) + (1-\alpha)N(k) \tag{5}$$

其中，k代表頻率，l代表音框，從(5)式可看出，$N(k)$的更新，除了有現階段$N(k)$的資訊外，也包含了上一個音框的$N(k)$資訊，而此權重 α 可依照環境自行調整。

2.2 聲源方位估測系統

2.2.1 陣列訊號處理

訊號源本身可以被清楚的分辨為某個頻率，稱之為窄頻訊號源(narrowband source)。相對於此，一個訊號是由一段頻帶內的頻率所組成，則稱之為寬頻訊號源(broadband/wideband signal source)。從物理角度來看，聲源可能離陣列麥克風相當遠，也有可能就在麥克風的附近。前者，是屬於遠場的情況(far-field)，陣列麥克風收到訊號源傳來的訊號，是屬於

平行波的形式；而後者是近場的情況(near-field)，陣列麥克風收到的訊號形式，是屬於球形波的形式。本論文中所要處理的聲源，是假設屬於遠場的窄頻訊號。

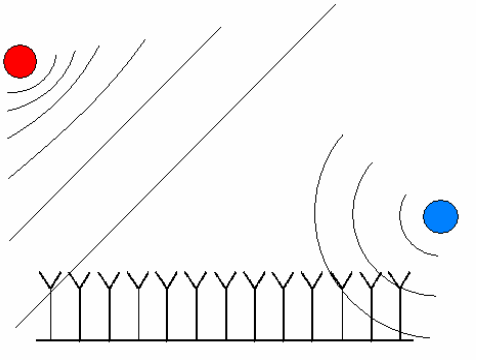


圖 2-2 遠場、近場示意圖

訊號之間的關係，可能是不相關(uncorrelated)、部分相關(partially correlated)、或是完全相關(completely coherent)。在自然界中，雜訊與人為的訊號，往往是不相關的，所以我們在麥克風訊號的處理上，也做了這個假設 [8]。

2.2.2 環形陣列麥克風

一般來說，我們最常用的陣列麥克風，可分為幾種排列方式：均勻線性排列麥克風(Uniform Linear Array)、平面排列麥克風(Planar Array)以及環形排列麥克風(Circular Array)。

均勻線性排列麥克風(Uniform Linear Array)，是最常使用的陣列麥克風形式，它的優點在於容易實現，且數學模型的推導最為簡單。以下是簡單的推導：

X1、X2、X3：代表麥克風收到的訊號

S：聲源

r：聲源到陣列麥克風的距離

Vc：波速

d：麥克風之間的固定距離

θ：訊號入射角度

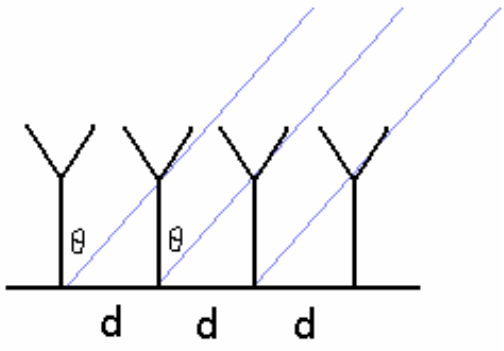


圖 2-3 均勻線性排列麥克風

$$\begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix} = \begin{bmatrix} S(t-\frac{r}{v_c}) \\ S(t-\frac{r}{v_c}-\frac{d\sin\theta}{v_c}) \\ S(t-\frac{r}{v_c}-\frac{2d\sin\theta}{v_c}) \end{bmatrix} \Leftrightarrow \begin{bmatrix} x_1(e^{jw}) \\ x_2(e^{jw}) \\ x_3(e^{jw}) \end{bmatrix} = \begin{bmatrix} S(e^{jw})e^{-jw\frac{r}{v_c}} \\ S(e^{jw})e^{-jw(\frac{r}{v_c}+\frac{d\sin\theta}{v_c})} \\ S(e^{jw})e^{-jw(\frac{r}{v_c}+\frac{2d\sin\theta}{v_c})} \end{bmatrix}$$

我們以第一顆麥克風作為參考點，則接收到的訊號在時域與頻域分別如以上的表示。以下我們做了兩個假設：一個是遠場的情況(far-field)，所以 r 將趨近於無限大；另一個是以第一顆麥克風接收到聲源的訊號，當作參考訊號。經過兩個假設，表示式簡化為以下形式：

$$\begin{bmatrix} x_1(e^{jw}) \\ x_2(e^{jw}) \\ x_3(e^{jw}) \end{bmatrix} = \begin{bmatrix} x_1(e^{jw}) \\ x_1(e^{jw})e^{-jw(\frac{d\sin\theta}{v_{s1}})} \\ x_1(e^{jw})e^{-jw(\frac{2d\sin\theta}{v_{s1}})} \end{bmatrix} = \begin{bmatrix} 1 \\ e^{-jw(\frac{d\sin\theta}{v_{s1}})} \\ e^{-jw(\frac{2d\sin\theta}{v_{s1}})} \end{bmatrix} \left[S_{x_1}(e^{jw}) \right] \tag{6}$$

以相同的道理，當聲源有三個的時候，則表示式可擴充為以下形式：

$$\begin{bmatrix} x_1(e^{jw}) \\ x_2(e^{jw}) \\ x_3(e^{jw}) \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ e^{-jw(\frac{d\sin\theta_1}{v_{s1}})} & e^{-jw(\frac{d\sin\theta_2}{v_{s2}})} & e^{-jw(\frac{d\sin\theta_3}{v_{s3}})} \\ e^{-jw(\frac{2d\sin\theta_1}{v_{s1}})} & e^{-jw(\frac{2d\sin\theta_2}{v_{s2}})} & e^{-jw(\frac{2d\sin\theta_3}{v_{s3}})} \end{bmatrix} \begin{bmatrix} S_{1x_1}(e^{jw}) \\ S_{2x_1}(e^{jw}) \\ S_{3x_1}(e^{jw}) \end{bmatrix} \tag{7}$$

以上的式子說明了麥克風端所接收到的訊號，是聲源經過空間中不同的延遲效應之後，疊加而成。而這個特殊的轉換矩陣，稱之為 manifold matrix [9]。

環形排列麥克風是改良均勻線性陣列的一種，它使用了球形座標、圓形的排列，我們改以圓心作為參考點，整個圓上的麥克風數目共有 M 個，推導得到的 manifold vector 為以下的形式：

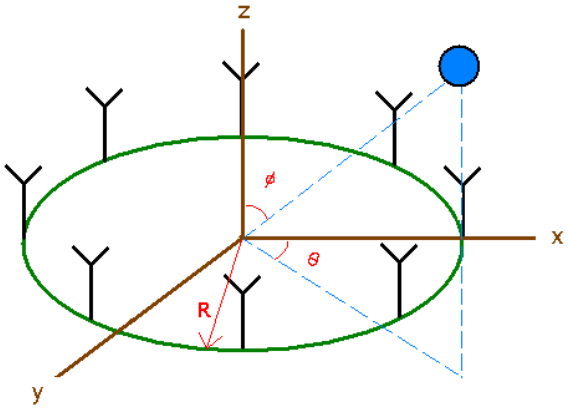


圖 2-4 平面環形陣列

$$a^T(\theta,\phi)=\left[\begin{array}{ccccc} e^{-jw\frac{R\sin\phi\cos\theta}{Vc}} & e^{-jw\frac{R\sin\phi\cos(\theta-\frac{2\pi}{M})}{Vc}} & \cdots & e^{-jw\frac{R\sin\phi\cos(\theta-\frac{2(M-1)\pi}{M})}{Vc}} \end{array}\right] \tag{8}$$

2.2.3 聲源方位估測

本論文使用 MUSIC 演算法對聲源角度估測，MUSIC 的全名是 Multiple Signals Classification Method。以下是這種方法的推導：

在陣列麥克風接收端收到的訊號中，除了訊號源發射的訊號以外，還混雜著雜訊在裡面，所以收到的訊號可以表示成以下的形式：

$$\begin{aligned} X(n) &= \sum_{i=1}^{M} a(\theta_i)S_i(n)+N(n) \\ &= AS_i(n)+N(n) \end{aligned} \tag{9}$$

其中 A 矩陣，就是先前推導出來的 manifold matrix。在訊號處理上，想針對一段訊號有更清楚的了解，最常用的方法就是去分析協方差(covariance)。首先，我們針對接收到的陣列訊號，做 auto-covariance：

$$\begin{aligned} R_{XX} &= E\left(x,x^H\right)=E\left(AS+N,(AS+N)^H\right) \\ &= AR_{SS}A^H+\sigma_N^2 I \end{aligned} \tag{10}$$

其中，訊號與雜訊 cross-variance 是零的原因，是基於前面的基本假設。因為雜訊的能量遠小於訊號，所以以上的式子可以簡化為

$$R_{xx} \approx AR_{SS}A^H$$

$$\Rightarrow\ V\Lambda^2V^H = AR_{SS}A^H$$

$$\Rightarrow\ [V_1\,V_2 \ldots V_M]\begin{bmatrix} \lambda_1^2 & 0 & \ldots & 0 \\ 0 & \lambda_2^2 & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots \\ 0 & 0 & \ldots & \lambda_M^2 \end{bmatrix}[V_1\,V_2 \ldots V_M]^H = AR_{SS}A^H \qquad (11)$$

在我們對訊號的 covarıance matrıx 做特徵值分解之後，我們就可以更深入到訊號空間來分析。首先，R_{ss} 與 R_{xx} 的關係，是透過 manifold matrix 的相似轉換，所以，向量空間 V_i 與 manifold matrix 的行向量，是對應到相同的空間。λ_1 到 λ_M 分別對應 V_1 到 V_M 的向量空間，從能量的觀點來看，訊號的能量會遠大於雜訊，所以在陣列天線數目大於訊號源的數目時，從 λ_1^2 到 λ_M^2 的大小，就可以判斷出何者為訊號，何者為雜訊。MUSIC 演算法另一個基本假設，是訊號源彼此之間是不相關(uncorrelated)，所以不同訊號源對應的訊號空間 V_i 彼此之間是互相正交(orthogonal)。藉由此正交的關係，我們可以由雜訊空間來反求出訊號的方向來 [9][10]。

$$\min(\|V_{noise}A(\theta)\|) \quad , \quad \theta = -180^\circ \sim 180^\circ \qquad (12)$$

2.3 語音純化系統

2.3.1 適應性濾波器簡介

通常而言，濾波器的係數通常設計出來後皆為固定的，並不會自動的變動。而適應性濾波器指的是能根據輸入信號，用訊號處理的技巧來適應性地調整濾波器係數，讓濾波效果更能適應現在環境，以完成某些特定的需要。

2.3.2 適應性濾波器處理架構

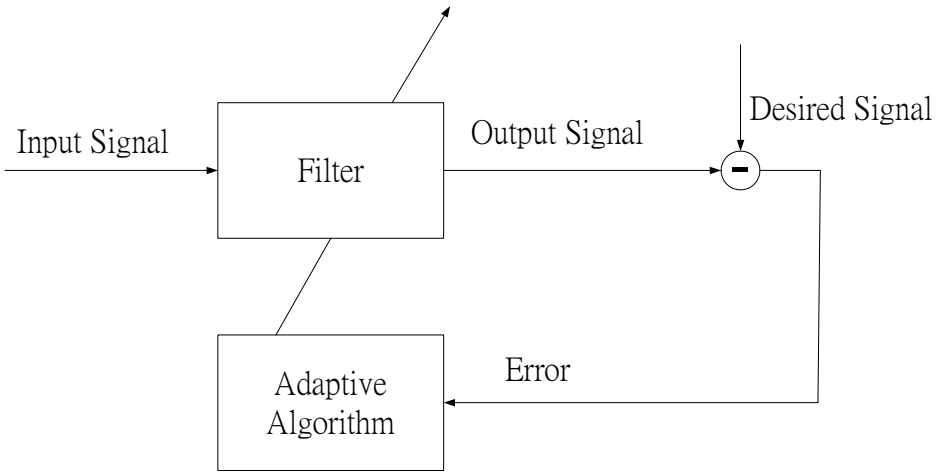


圖 2-5 適應性濾波器處理架構圖

適應性濾波器處理架構圖如圖 2-5 所示，當輸入訊號經過濾波器處理後，會與希望達成的訊號相減而產生誤差訊號，此誤差訊號經過適應性演算法的運算而調整濾波器的係數，使誤差訊號降低。如此反覆運算會讓濾波器係數不斷地變動，讓濾波器輸出訊號與希望達成的訊號愈來愈相近。

2.3.3 結合真人語音活動偵測與適應性空間濾波器之架構

語音純化系統其架構圖[11]如圖 2-6 所示，在圖中，Upper Beamformer 與 Lower Beamformer 皆為空間濾波器，而 Beamformer 的數學表示式如下:

$$y(n)=\sum_{i=1}^{M} w_{i}^{H} x_{i}(n) = \mathbf{w}^{\mathrm{H}}\mathbf{x}(\mathrm{n}) \quad \mathrm{n} = 1,2\ldots\ldots$$

在實際應用上，**x** (n)為輸入的麥克風聲音訊號，M 為麥克風數目，**w** 為空間濾波器係數，而 y(n)即為空間濾波器的輸出，因此 Beamformer 會對輸入的多通道聲音訊號做乘上權重然後疊加的動作。真人語音活動偵測用於 Lower Beamformer 後，因此麥克風陣列訊號皆會先經過 Lower Beamformer，再通過語音活動判定，不論語音活動判定結果如何，系統都會將通過 Lower Beamformer 後的結果輸出，但若判定為非真人語音，系統會將收到的非真人語音訊號（未通過 Lower Beamformer）加上參考訊號(預錄的無噪音語音資料)做相加並傳遞給 Upper Beamformer 做適應性訊號調

整，所謂的適應性訊號調整就是當輸入訊號經過 Upper Beamformer 處理後，會與希望達成的訊號相減而產生誤差訊號，此誤差訊號經過適應性演算法 Normalize Least-Mean-Square Algorithm 的運算來調整濾波器的係數，使誤差訊號降低。如此反覆運算會讓濾波器係數不斷地變動，讓濾波器輸出訊號與希望達成的訊號愈來愈相近。調整完畢後再將濾波器係數傳遞給 Lower Beamformer，更新 Lower Beamformer 濾波係數，使得 Lower Beamformer 具有抑制雜訊純化語音的功能。

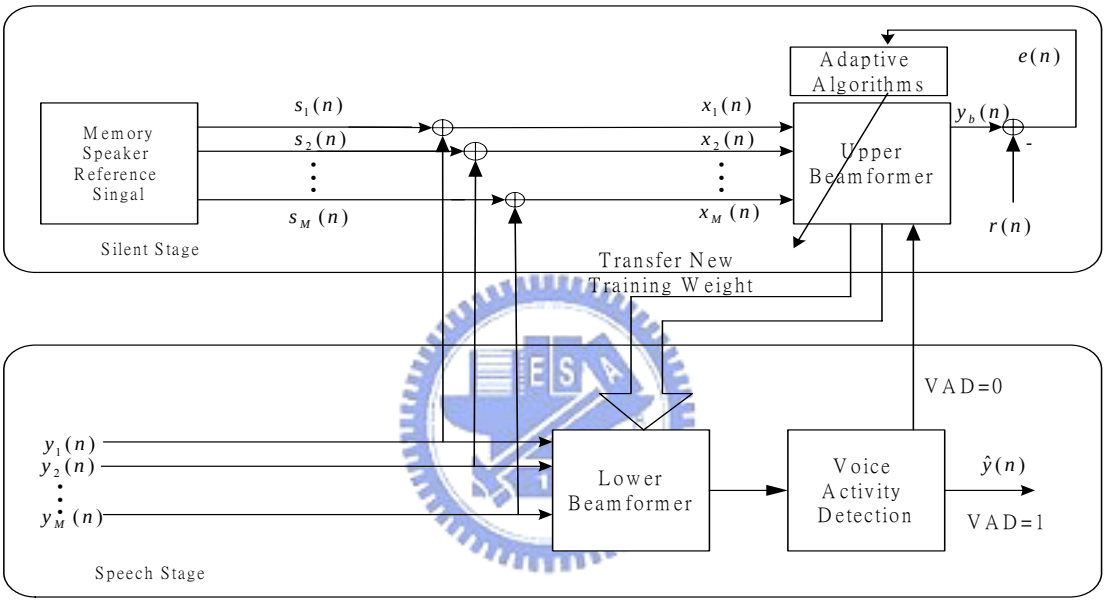


圖 2-6 即時適應性語音純化系統架構圖

2.3.4 Least-Mean-Square（LMS）Algorithm

LMS 演算法指的是，找出一組權重 W 使得誤差平方項最小[12]。其基本架構如圖 2-8 所示。假設希望達成的訊號為 zero-mean，其變異量為 σ_d^2

$E\{d\}=0, \sigma_d^2 = E|d|^2$

而輸入訊號 x 為一組 $M \times 1$ 向量，並定義其共異量矩陣和互共異量矩陣

$R_x = E\{x^* x\}$ ， $R_{dx} = E\{dx^*\}$

因此目標函數如(13)式所示

$$J(w) \equiv \min_{w} E\{d - xw\}^2 = E(d - xw)(d - xw)^* \tag{13}$$

方程式(13)的意義就是找出一組 W 使誤差平方項最小，而 W 的找法則需用 Steepest-Descend Method，其標準式如下：

（new guess）=（old guess） ＋ （a correction term）

也就是

$$w_i = w_{i-1} + \mu p, \qquad i \geq 0 \tag{14}$$

其中(14)式意義為從 w_{i-1} 出發，並前進 μp 的距離，μ 為一個比重稱為 stepsize。而 p 的選取必須從(13)式下手，將(13)式展開可得

$$J(w) = \sigma_d^2 - R_{dx}^* w - w^* R_{dx} + w^* R_x w \tag{15}$$

為了找組 W 使 $J(w)$ 最小，對(15)式取 ∇_w 得

$$\nabla_w J(w) = w^* R_x - R_{dx}^*$$

(16)因此，為了讓 w 往 $J(w)$ 最低處的方向與強度前進，我們取

$$p = -[\nabla_w J(w_{i-1})]^* = R_{dx} - R_x w_{i-1} \tag{17}$$

方程式(14)可寫為

$$w_i = w_{i-1} + \mu[R_{dx} - R_x w_{i-1}] \qquad i \geq 0 \tag{18}$$

在實做上，R_{dx} 和 R_x 可用離散形式近似於瞬間值：

$$R_{dx} = d(i)x^*(i) \qquad R_x = x^*(i)x(i) \tag{19}$$

所以(18)式可寫為：

$$w(i) = w(i-1) + \mu x^*(i)[d(i) - x(i)w(i-1)] \qquad i \geq 0 \tag{20}$$

因此，LMS Algorithm 可整理如下：

Filter out ： $y(i) = x(i)w(i)$ (21)

Error function： $e(i) = d(i) - y(i)$ (22)

Update weight： $w(i) = w(i-1) + \mu x^*(i)e(i) \qquad i \geq 0$ (23)

其方塊圖如圖 2-7 所示。

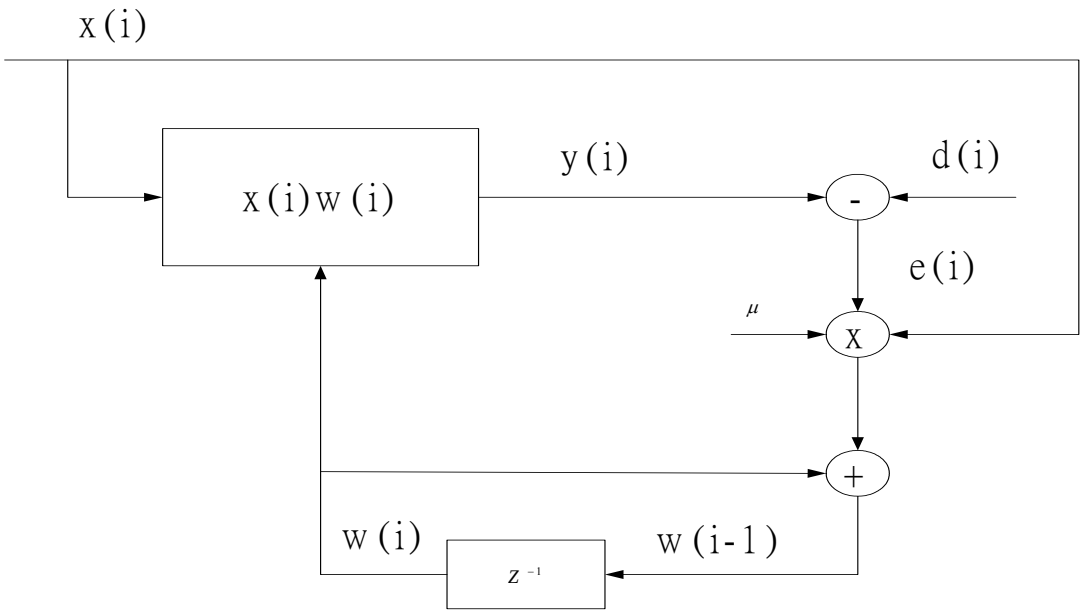


圖 2-7　LMS 演算法方塊圖

2.3.5 Normalize LMS Algorithm

在 LMS 演算法中，為了確保其收斂，μ 的範圍必須為 $0 < \mu < \frac{2}{\lambda_{\max}}$，$\lambda_{\max}$ 為 R_x 的最大特徵值，若所需濾波器階數愈高，則解 R_x 的特徵值就愈複雜，以實作方面來講，如此大的運算量會造成龐大的負擔，因此為了簡化其運算量，衍生出另一種演算法，Normalize LMS Algorithm[12]：

Filter out　：　$y(i) = x(i)w(i)$　(24)

Error function：　$e(i) = d(i) - y(i)$　(25)

Update weight：$w(i) = w(i-1) + \frac{\alpha x(i)e(i)}{\gamma + x^*(i)x(i)}$　$i \geq 0$　(26)

與 LMS 演算法比較，Normalize LMS 演算法只有在更新權重的部分不一樣，原有的 μ 被 $\frac{\alpha}{\gamma + x^*(i)x(i)}$ 所取代，其中，0<α<2，γ 為一個微小的數，目的只是確保分母項不為零，如此即可確保 Normalize LMS 演算法收斂，而且如此的運算即不用解 R_x 的特徵值，讓運算量降低許多。

2.4 人臉追蹤系統

平均位移追蹤演算法(Mean-shift)[14]是一種以樣本為基礎(template base)的常用的影像追蹤演算法，本人臉追蹤系統則是透過此演算法擷取人臉特徵，並對此特徵做即時且持續的追蹤。以下介紹此演算法原理。

演算法首先會將欲追蹤的目標的影像區域做擷取特徵的動作並建立成為目標模型(target model)，將掃描區域視為目標候選人(target candidate)。定義 $I_x = \{\mathbf{x}_i, u_i\}_{i=1,\ldots,M}$ 為目標模型影像，其中 M 為影像區域像素個數，i 代表第幾個像素，定義 $I_y = \{\mathbf{y}_j, v_j\}_{j=1,\ldots,N}$ 為目標候選人影像，其中 N 為影像區域像素個數，j 代表第幾個像素，而其中 $\mathbf{x}_i$ 與 $\mathbf{y}_j$ 分別為像素所在位置，我們將色彩空間分割為 B 個區間(bin)，u_i 與 v_j 則是代表該像素落在 0~B-1 的第幾個區間。

定義 *p(x,u)*為目標模型的機率特徵函數。使用高斯函數作為其核心函數(kernel function)，數學模型如下:

$$p(\mathbf{x},u) = \frac{1}{M}\sum_{i=1}^{M} \frac{\exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_{P,b(i)})^T(\Sigma_{P,b(i)})^{-1}(\mathbf{x}-\boldsymbol{\mu}_{P,b(i)})\right)}{2\pi\left|\Sigma_{P,b(i)}\right|^{1/2}}\delta[u-b(i)]$$

$$\triangleq \frac{1}{M}\sum_{i=1}^{M} K_P(\mathbf{x}-\boldsymbol{\mu}_{P,b(i)}, \Sigma_{P,b(i)})\delta[u-b(i)] \qquad (27)$$

其中 $b(i)$ 代表第 i 個像素所屬的色彩區間，K_P 為二維高斯核心函數，$\boldsymbol{\mu}_{P,b}$ 代表目標模型影像區域中屬於第 b 個色彩區間的像素的 mean vector，而 $\Sigma_{P,b}$ 則代表目標模型影像區域中屬於第 b 個色彩區間的像素的 covariance matrix，而 δ 為 Kronecher delta function。

接著定義相似度函數為:

$$J(I_x, I_y) = J(\mathbf{y}) = \frac{1}{N}\sum_{j=1}^{N} p(\mathbf{y}_j, v_j)$$

$$= \frac{1}{NM}\sum_{j=1}^{N}\sum_{i=1}^{M} K_P(\mathbf{y}_j - \boldsymbol{\mu}_{P,b(i)}, \Sigma_{P,b(i)})\delta[v_j - b(i)] \qquad (28)$$

相似度函數將候選模組影像各像素值帶入模型模組的數學模型中並作疊加計算其相似程度機率值，因此若兩者越相似則帶入的結果值越大。

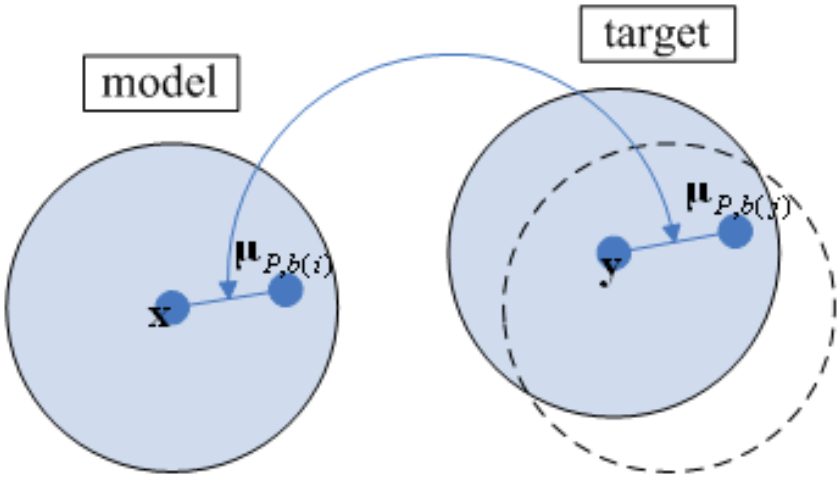


圖 2-8 理想影像移動示意圖

假設被追蹤目標在追蹤過程中色彩或形狀上沒有改變，只有單純的移動，如圖 2-8 所示，則第 b 個色彩區間的影像會保持以下關係:

$$\boldsymbol{\mu}_{P,b(i)} - \mathbf{x} = \boldsymbol{\mu}_{P,b(j)} - \mathbf{y} \tag{29}$$

$$\boldsymbol{\mu}_{P,b(i)} = \boldsymbol{\mu}_{P,b(j)} - \mathbf{y} + \mathbf{x} \tag{30}$$

將(30)式帶入(28)式中可得相似函數為:

$$J(\mathbf{y}) = \frac{1}{NM}\sum_{j=1}^{N}\sum_{i=1}^{M} K_P(\mathbf{y}_j - \boldsymbol{\mu}_{P,b(j)} + \mathbf{y} - \mathbf{x}, \Sigma_{P,b(i)})\delta[v_j - b(i)] \tag{31}$$

所以我們希望找到一個 $\mathbf{y}$ 向量使得 $J(\mathbf{y})$最大，也就是找到一個 $\mathbf{y}$ 向量使得 $J(\mathbf{y})$取梯度等於零。

$$\nabla J(\mathbf{y}) = \mathbf{0}$$

$$\Rightarrow \frac{1}{NM}\sum_{j=1}^{N}\sum_{i=1}^{M} -(\Sigma_{P,b(i)})^{-1}(\mathbf{y}_j - \boldsymbol{\mu}_{P,b(j)} + \mathbf{y} - \mathbf{x})K_P\delta[v_j - b(i)] = \mathbf{0}$$

$$\Rightarrow \left\{\sum_{j=1}^{N}\sum_{i=1}^{M}(\Sigma_{P,b(i)})^{-1}K_P\delta[v_j - b(i)]\right\}(\mathbf{y} - \mathbf{x}) = \sum_{j=1}^{N}\sum_{i=1}^{M}(\Sigma_{P,b(i)})^{-1}(\mathbf{y}_j - \boldsymbol{\mu}_{P,b(j)})K_P\delta[v_j - b(i)]$$

$$\mathbf{y} - \mathbf{x} = \left\{\sum_{j=1}^{N}\sum_{i=1}^{M}(\Sigma_{P,b(i)})^{-1}K_P\delta[v_j - b(i)]\right\}^{-1}\left\{\sum_{j=1}^{N}\sum_{i=1}^{M}(\Sigma_{P,b(i)})^{-1}(\mathbf{y}_j - \boldsymbol{\mu}_{P,b(j)})K_P\delta[v_j - b(i)]\right\} \tag{32}$$

最後可以得知追蹤的目標的新所在位置 $\mathbf{y}_{new}$ 為:

$$\mathbf{y}_{new}=\left\{\sum_{j=1}^{N}\sum_{i=1}^{M}(\Sigma_{P,b(i)})^{-1}K_P\delta[v_j-b(i)]\right\}^{-1}\left\{\sum_{j=1}^{N}\sum_{i=1}^{M}(\Sigma_{P,b(i)})^{-1}(\mathbf{y}_j-\boldsymbol{\mu}_{P,b(j)})K_P\delta[v_j-b(i)]\right\}+\mathbf{x} \tag{33}$$

其中

$$K_P(\mathbf{y}_j-\boldsymbol{\mu}_{P,b(j)}+\mathbf{y}_{old}-\mathbf{x},\Sigma_{P,b(i)}) = \frac{\exp\left(-\frac{1}{2}(\mathbf{y}_j-\boldsymbol{\mu}_{P,b(j)}+\mathbf{y}_{old}-\mathbf{x})^T(\Sigma_{P,b(i)})^{-1}(\mathbf{y}_j-\boldsymbol{\mu}_{P,b(j)}+\mathbf{y}_{old}-\mathbf{x})\right)}{2\pi\left|\Sigma_{P,b(i)}\right|^{1/2}} \tag{34}$$

以下為平均位移演算法流程圖:

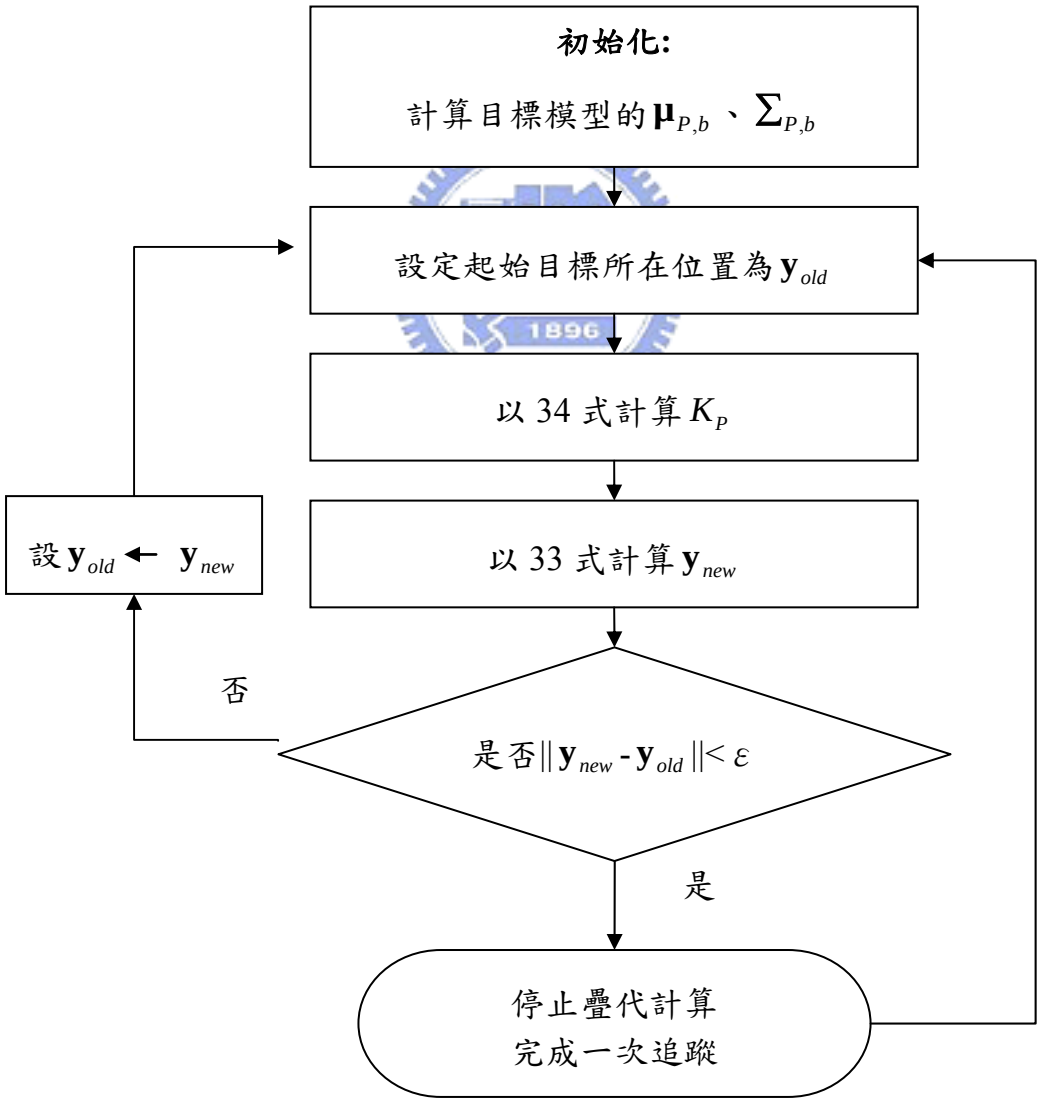


圖 2-9 平均位移演算法流程圖

第三章　系統軟硬體設計與實現

3.1 硬體環境

本系統的實際成品圖如下所示，硬體上包括了環型數位式麥克風陣列、FPGA、DM6446 平台(DaVinci)、PTZ 攝影機與顯示器。以下小節會對其分別介紹。

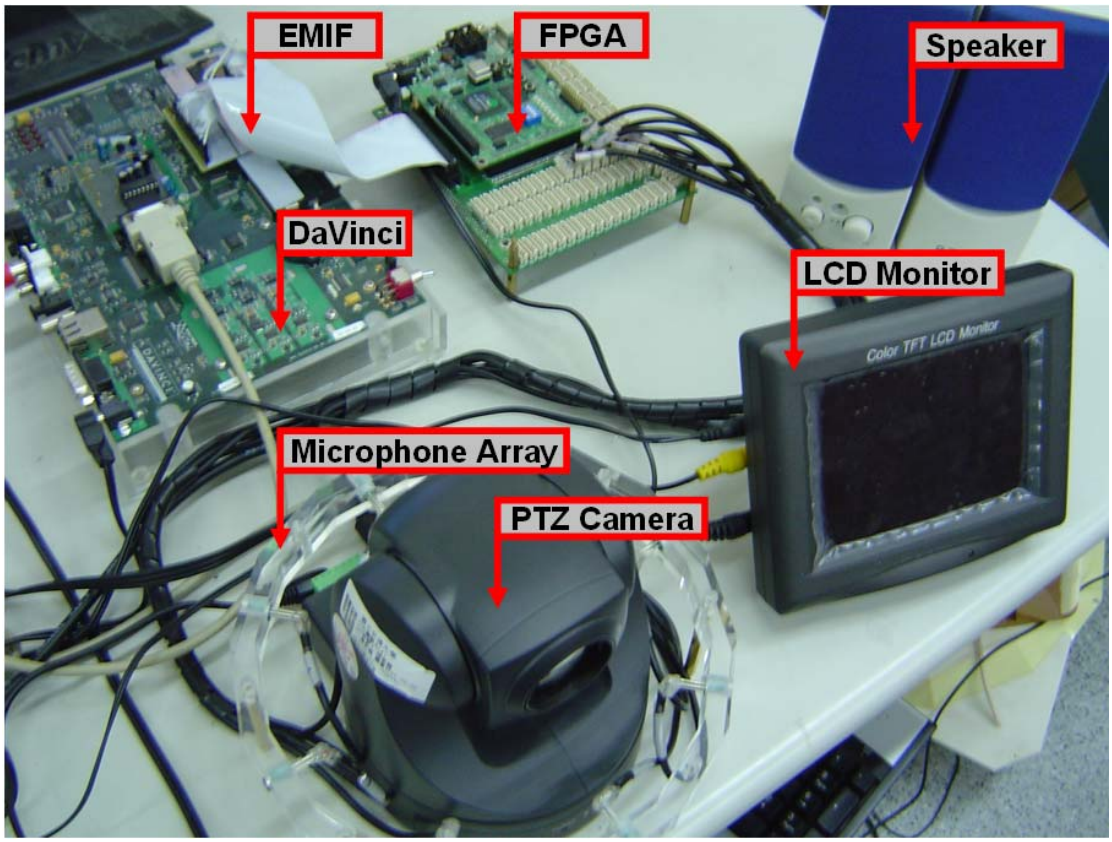


圖 3-1　即時影音追蹤與語音純化系統硬體圖

3.1.1 雙核心發展平台 DM6446 EVM

整體系統以 TI 所發行的 DM6446 EVM 為發展平台。平台的 DM6446 SoC (DaVinci)主要包含了兩個運算核心，具有一個適合一般計算與系統控制的 ARM RISC CPU 與一個適合處理影音演算法運算的 DSP，DM6446 SoC 架構[15][16]如圖 3-2 所示:

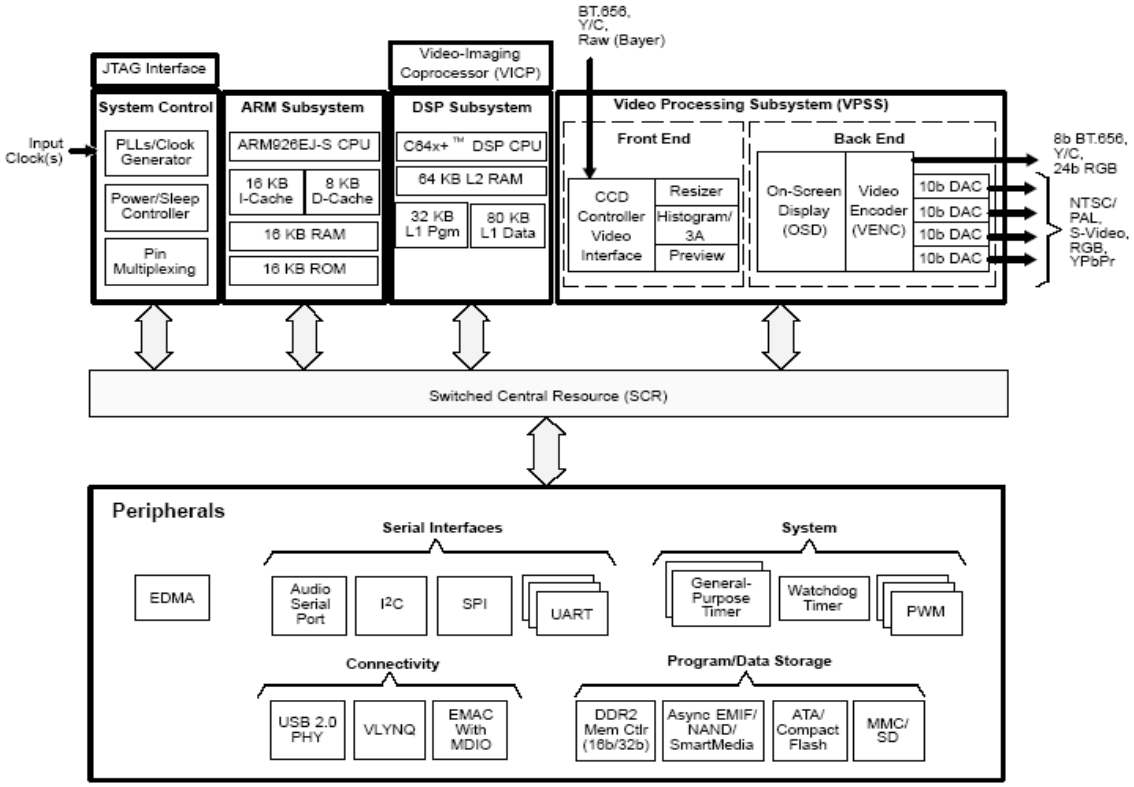


圖 3-2　DM6446 SoC 架構圖

整個SoC架構大致可以切分為三個子系統(Subsystem)與周邊共四大區塊。ARM 子系統包含了ARM926 RISC 核心與其相關記憶體，而DSP子系統包含C64x+ DSP、影像協處理器(Video-Imaging Coprocessor)與其相關記憶體。此外DM6446具有影像處理子系統，可對輸入影像以硬體做前處理動作並對輸出影像以硬體做後處理動作，因此使用影像處理子系統可降低系統在影像處理上的運算負擔。DM6446的眾多周邊中，我們選用非同步外部記憶體介面(AEMIF, Asynchronous External Memory Interface)與數位式麥克風陣列聲音訊號擷取系統溝通做為資料傳遞介面，其詳細溝通方式會在3.5節的多通道聲音擷取架構加以介紹。圖3-3為DM6446 EVM發展平台的外觀。而圖3-4為其對應的硬體方塊圖，圖3-4中DC1即為用來跟數位式麥克風陣列聲音訊號擷取系統溝通的非同步外部記憶體介面。

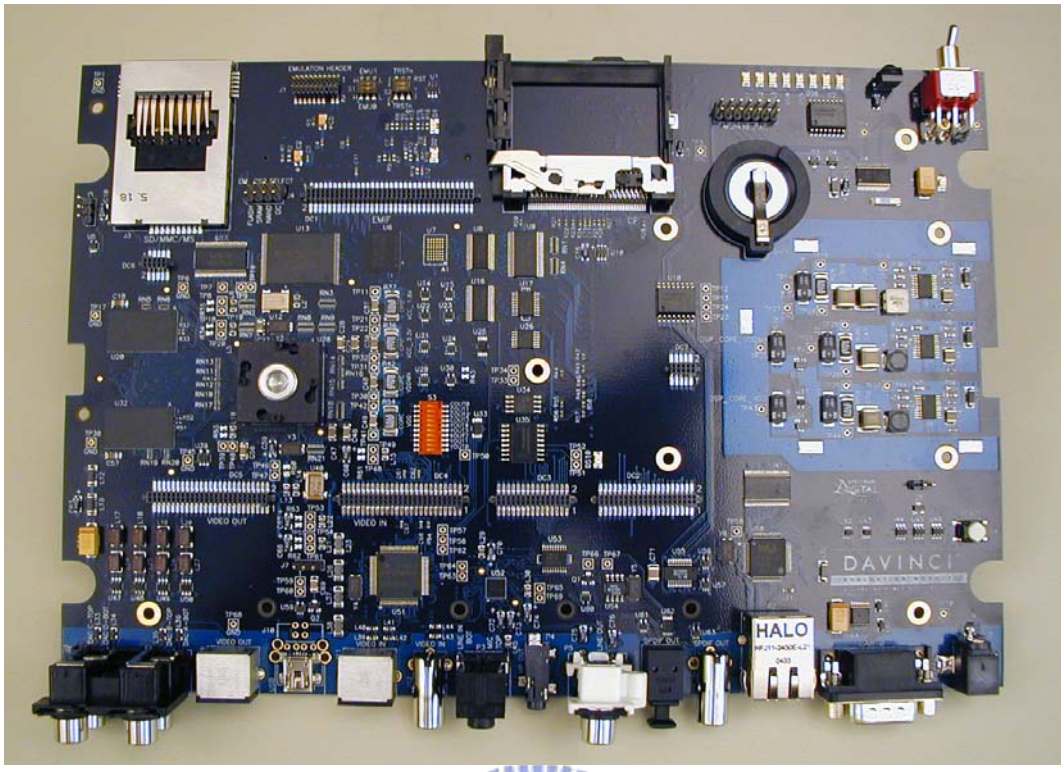


圖 3-3　DM6446 EVM 外觀

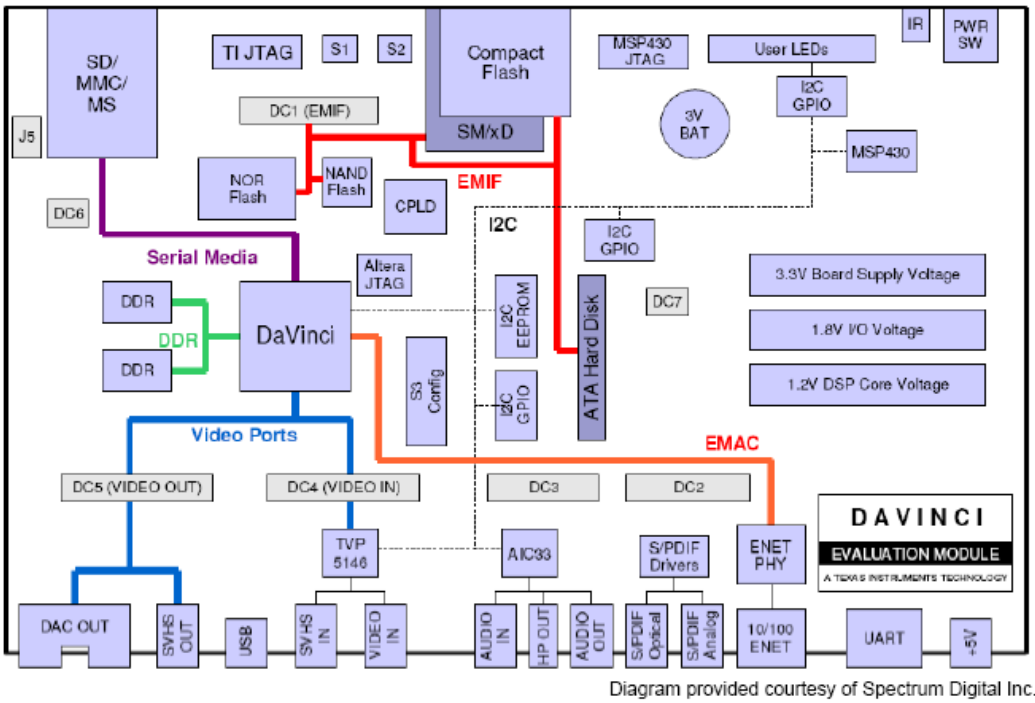


圖 3-4 硬體方塊圖

以下列出較為重要的DM6446 EVM 硬體規格。

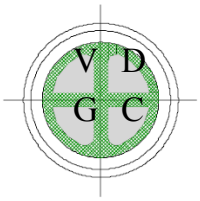
- DM644x processor with an ARM processor operating up to 300 MHz. and a C64xx DSP operating up to 600 MHz
- 1 video input port, supports composite or S video
- 4 video DAC outputs - component, RGB, composite
- 256 Mbytes of DDR2 DRAM
- 16 Mbytes NOR Flash, 64 Mbytes NAND Flash, 4Mbytes SRAM
- AIC33 stereo codec
- USB2 Interface
- 10/100 MBS Ethernet Interface

3.1.2 數位式麥克風陣列聲音訊號擷取系統

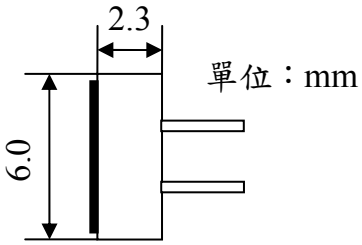
數位式麥克風陣列聲音訊號擷取系統主要是由數位式麥克風、壓克力環形陣列、連接 IO 板與 FPGA 共四部份硬體所構成，以下分別介紹:

- **數位式麥克風**

數位式麥克風為本實驗室自行研發設計，發包給音賜公司進行量產。架構圖見圖 3-5。圖中可見數位麥克風有四根腳位，其中時脈使用 1.2 MHz，輸出 1-bit 的數位訊號。



V：VCC
D：Delta Sigma
G：GND
C：CLK



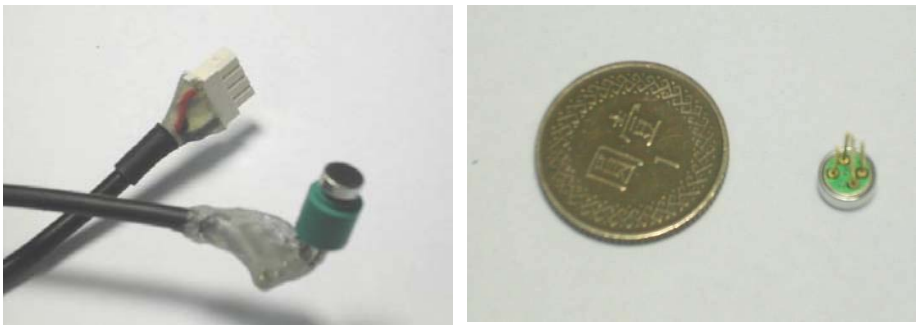


圖 3-5 數位麥克風架構與實際成品圖

類比式麥克風經過聲音震動產生電壓後，需要通過放大器及 A/D 轉換器轉成數位資料。然而隨著麥克風個數增加，放大電路的體積也跟著增加，且多個通道同時取樣的 A/D 轉換器價格也越貴。相對於類比式麥克風，數位式麥克風已將放大器與 A/D 轉換器電路內嵌至麥克風中，因此輸出已為數位訊號。

■ 壓克力環形陣列

圖 3-6 為壓克力環形陣列的設計圖，陣列內接圓半徑為 90mm，環上一共可鑲嵌 8 顆數位式麥克風，每顆數位式麥克風間夾 45 度角。而圖 3-7 為壓克力環形陣列，鑲嵌數位式麥克風後的實體圖。

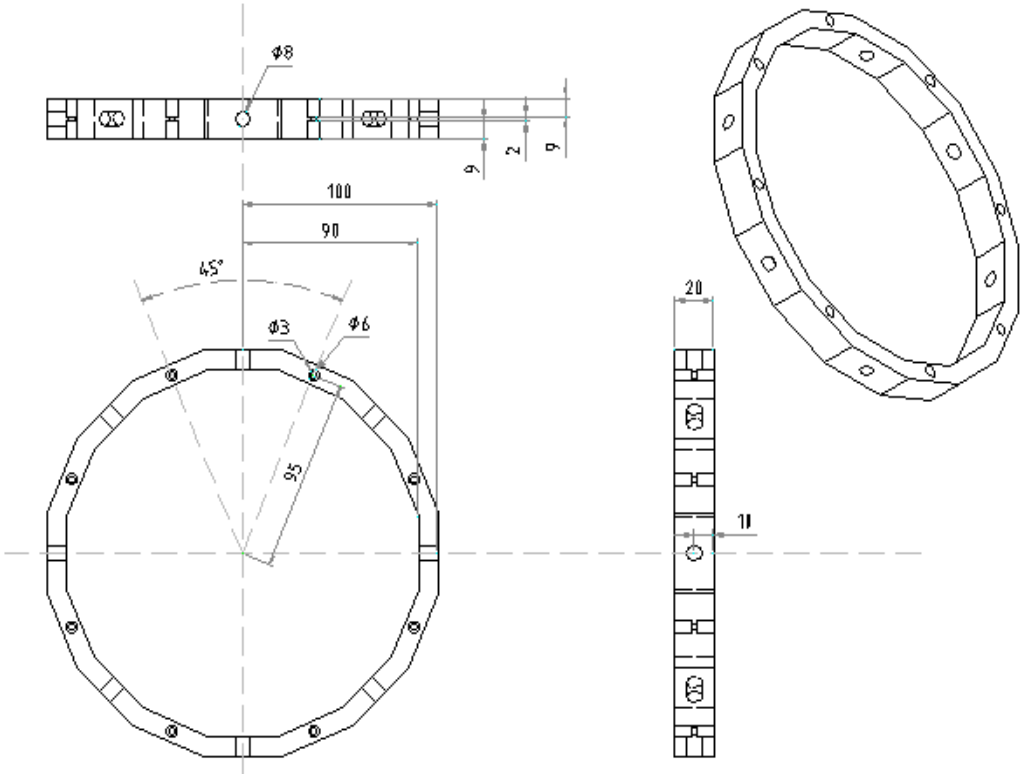


圖 3-6 壓克力環形陣列的設計圖

圖 3-7 為環形數位式麥克風陣列實體圖。

■ **數位式麥克風陣列連接 IO 板**

本實驗室自行設計一套 I/O 板以供數位麥克風陣列作應用。此板提供 76 個數位式麥克風插槽，I/O 板中間部分連接 FPGA 單板，透過 FPGA 程式撰寫，可與不同介面平台作溝通。I/O 單板下方的腳位可與 DM6446 EVM 的非同步外部記憶體介面透過排線連結做資料的傳輸。I/O 單板背面提供三種 regulator 的介面，將 5V 轉成 1.8V 供數位式麥克風陣列使用。

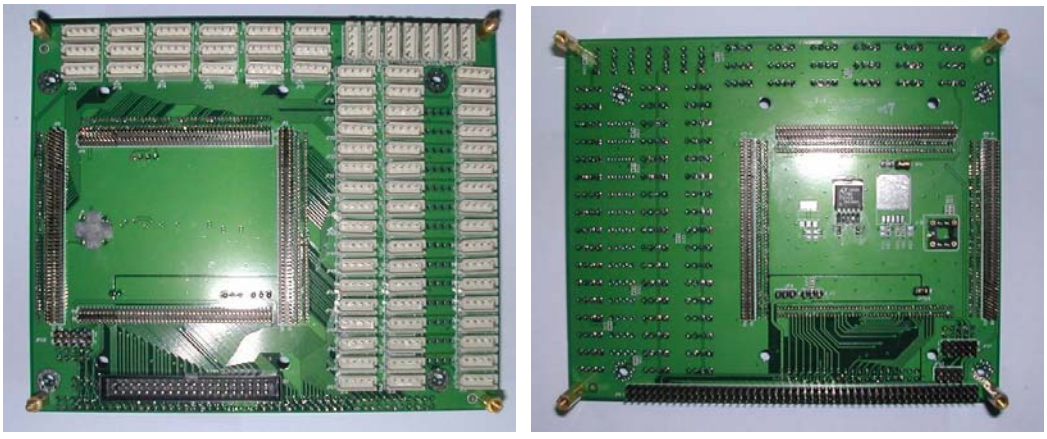


圖 3-8 I/O 板實際成品(左圖正面，右圖反面)

■ **FPGA 硬體**

FPGA 我們使用的是 ALTERA Cyclone II 系列的 EP2C35F484C6N 晶片，由茂綸公司所開發的實驗板。尺寸為 11 cm×8 cm。如圖 3-9。

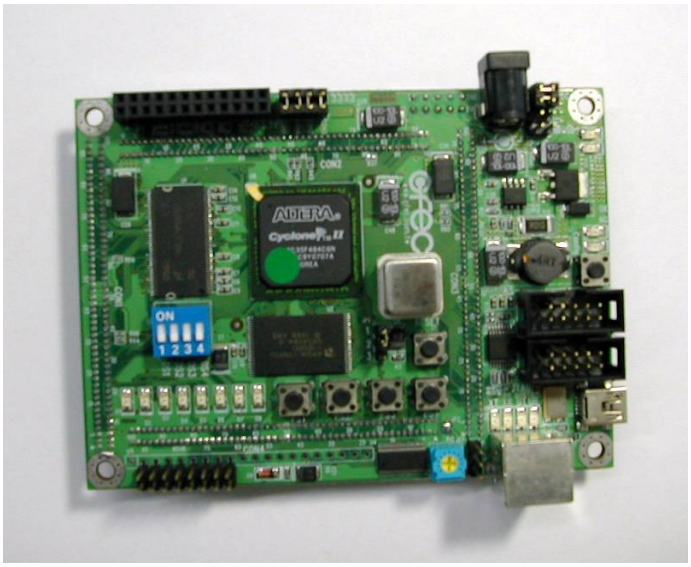


圖 3-9　GFEC Cyclone II Strarter Kit

3.1.3 系統其他硬體周邊

以下介紹系統使用到的其他相關周邊硬體。

■ **PTZ 攝影機**

圖 3-10 為系統擷取影像所使用到的 SONY EVI-D70 PTZ 攝影機，NTSC 影像格式，可透過 RS232 對其作伺服馬達控制，水平旋轉正負 170 度，上下移動正負 120 度。

圖 3-10　SONY EVI-D70 PTZ 攝影機

■ **彩色 TFT LCD 螢幕**

圖 3-11 為系統影像輸出所使用到的 EverFocus EN220 5.6 吋彩色 TFT LCD 螢幕，為 NTSC 影像格式。

圖 3-11 彩色 TFT LCD 螢幕

■ **JTAG 仿真器與 JTAG 轉接板**

圖 3-12 為 XDS510USBJTAG 仿真器(emulator)，XDS510USBJTAG 仿真器所擔任的角色就是作為介於開發程式軟體(CCS)以及 DSP 間的橋樑，控制 DSP 和 CCS 間的資料傳輸及交換，CCS 可藉由 XDS510USBJTAG 仿真器再透過 DSP 的 JTAG 埠控制 DSP 的執行並將 DSP 內的資料即時傳出用以偵錯。圖 3-13 為 JTAG 轉接板，此轉接板是用於 XDS510USBJTAG 仿真器與 DM6446 間，轉換兩者間的 PIN 腳數 (20/14)、電壓與時脈。開發程式軟體(CCS)會在 3.3 軟體環境章節介紹。

圖 3-12 XDS510USBJTAG仿真器

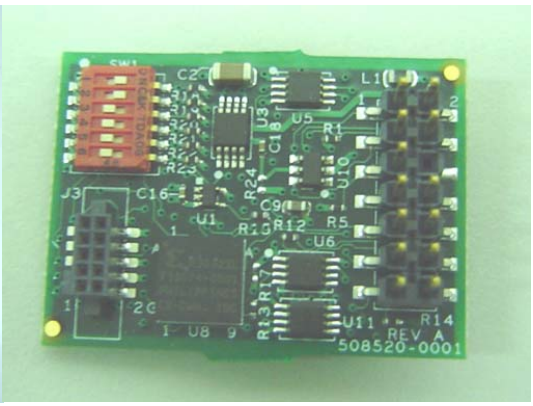


圖 3-13 JTAG轉接板

3.2 硬體系統架構

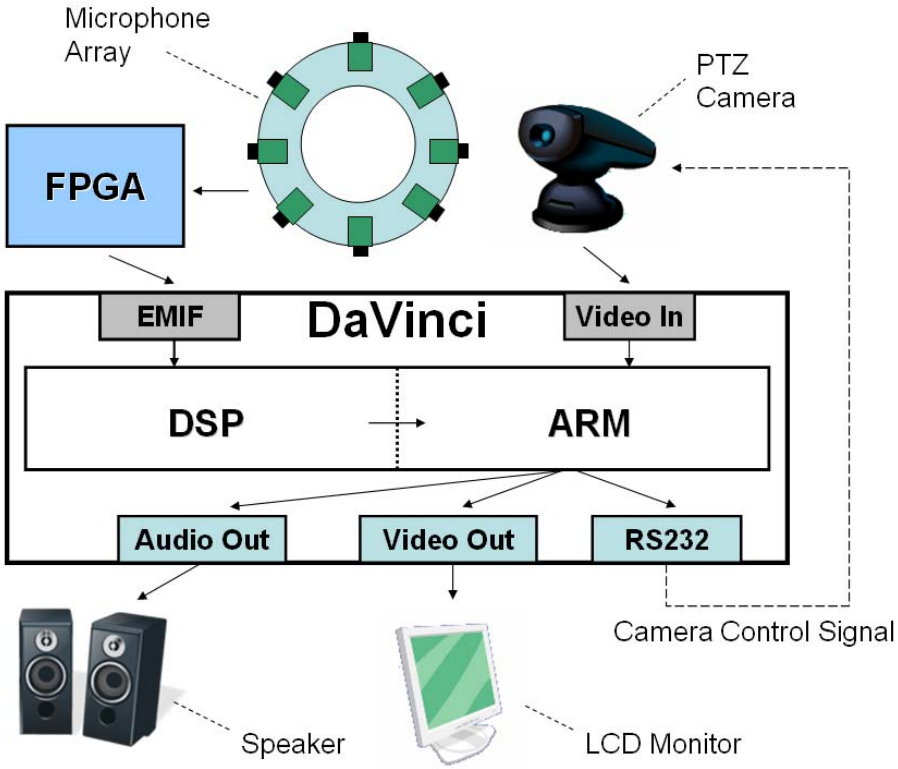


圖 3-14 硬體系統架構圖

本系統可以切分成影像與聲音兩個子系統，聲音子系統的聲音資料擷取是透過數位式麥克風陣列訊號擷取系統擷取，數位式麥克風陣列訊號擷取系統主要由八顆數位式麥克風所排成的環形陣列與一塊 FPGA 所組成，數位式麥克風收音後，輸出的資料格式為 sigma-delta 格式，因此麥克風的輸出需要透過抽樣濾波器(decimation filter)將 sigma-delta 資料格式轉換為 16-bits 資料格式，抽樣濾波器包含了降頻器、低通濾波器與高通濾波器，用以濾除高頻及低頻雜訊，並使取樣訊號符合 Nyquist-Shannon Sampling Theorem。當抽樣濾波器輸出 16-bits 資料時，平台需要一個多工器將多頻道的資料輪流傳出，此多工器與抽樣濾波器皆規劃於 FPGA 上實現。八顆數位式麥克風的聲音訊號在 FPGA 上做前處理後，FPGA 會透過 DM6446 EVM 的非同步外部記憶體介面(AEMIF, Asynchronous External Memory

Interface) 將處理後的聲音訊號傳輸到後端的 DM6446，因此 FPGA 會將聲音資料包裝成非同步外部記憶體介面適用的格式傳出，而 DM6446 會將 FPGA 視為一塊外部記憶體來抓取資料。DM6446 透過非同步外部記憶體介面抓取聲音資料並執行語音活動偵測、聲源方位估測與適應性語音純化的動作，經由估測出的聲源方位資訊會透過 RS232 串列埠對 PTZ 攝影機做伺服控制，可控制攝影機鏡頭對準發聲源，以輔助影像子系統解決影像脫鎖和對未進入 CCD 鏡頭前對影像位置之估測問題。而純化後的語音資料則是透過 AUDIO OUT 作類比輸出。

影像子系統的影像資訊透過 PTZ 攝影機擷取，影像資訊直接透過 DM6446 的影像介面輸入，DM6446 透過擷取的影像即時追蹤人臉，並將經過人臉追蹤系統處理後的影像透過 DM6446 的影像輸出介面送至 TFT LCD 螢幕顯示。

3.3 軟體環境

3.3.1 Linux 作業系統與編譯器

一般來說嵌入式平台的硬體資源較PC拮据，平台處理器運算速度與記憶體大小都會受到限制，因此程式開發者通常不會直接於嵌入式系統板上開發程式，而會選擇額外使用一台PC作為嵌入式系統程式開發的環境。我們使用TI 提供的MontaVista Linux V4.0做為DM6446平台ARM端的嵌入式作業系統，而PC上則是安裝Red Hat Enterprise Linux V4作業系統作為平台的程式開發環境，此外PC的Linux上會安裝MontaVista Linux V4.0 System Tools，內含相關系統開發套件，包括了編譯器與平台的Linux kernel與file system的重製工具。由於我們使用x86架構的PC作為開發平台，因此不能直接使用其GCC作為編譯器，而是使用可編譯出ARM端指令集的轉換編譯器(Cross-compiler)來編譯程式。我們使用arm_v5t_le-gcc 作為轉換編譯器。

3.3.2 Code Composer Studio

Code Composer Studio (CCS)為 TI 提供的一套整合式開發環境的發展軟體，其中提供了 C 語言的編譯器、組合語言的組譯器與其他相關工具。ARM 端程式可以在 Linux 環境下開發，但 DSP 端程式則是需要透過 CCS 開發會較為便利，使用 CCS 可以快速簡單的中斷程式並觀察記憶體內部各暫存器數值，以便於 DSP 程式開發除錯。使用 CCS 需搭配使用在硬體環境章節已介紹的 JTAG 仿真器使用。下圖 3-15 即為 CCS 的整合開發介面。

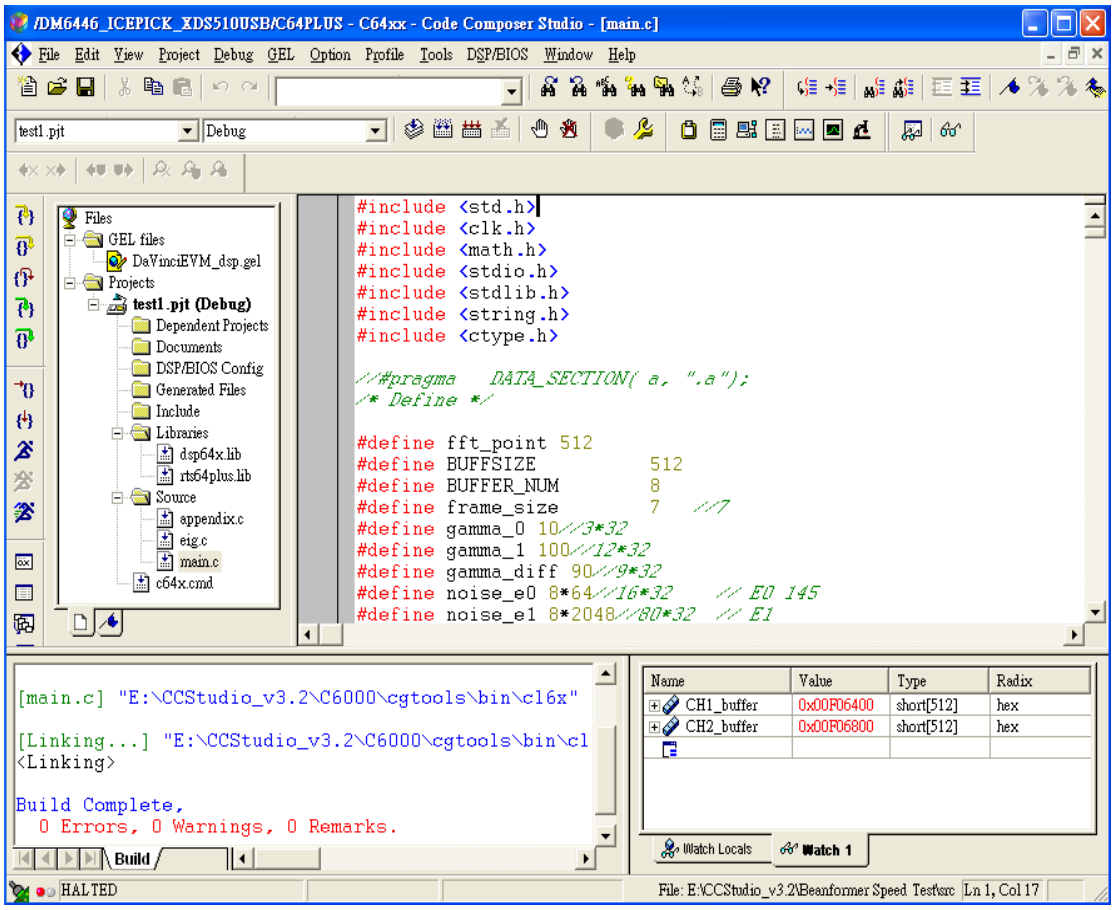


圖 3-15　CCS 的整合開發介面

3.3.3 DSP/BIOS Link

DSP/BIOS Link[17]是一套TI提供用來處理雙核心系統平台內部的GPP (General Propose CPU)與DSP間溝通的軟體，在DM6446平台上的GPP即為ARM處理器，DSP/BIOS Link這套IPC(Inter – Processor Communication)軟

體提供了一些通用API，幫助雙核心程式開發者避開一些底層較複雜的溝通協定機制，可直接專注於程式應用面發展。我們使用的DSP/BIOS LINK軟體版本為v1.30.06 。

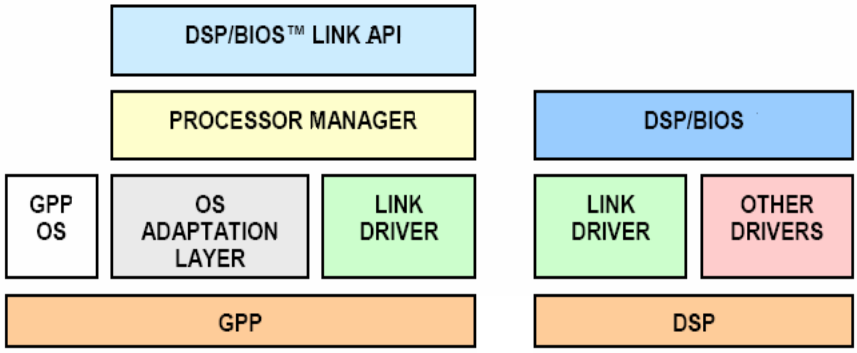


圖 3-16　DSP/BIOS LINK 軟體架構圖

圖 3-16 為 DSP/BIOS LINK 的軟體架構圖，各方塊功能如下:

- **GPP 端 :**
  - **OS ADAPTATION LAYER :**

    OS ADAPTATION LAYER 將 GPP OS 包在其中，使 OS 與其他元件分離，只拉出幾個通用 API 跟其他元件溝通，其他元件都得使用這些 API 而不能直接呼叫 OS，如此可以讓 DSP/BIOS LINK 軟體工作在不同 OS 下。
  - **LINK DRIVER :**

    LINK DRIVER 把低階的控制作業包在其中，負責控制著 DSP 與使用著已定義的低階控制協定在 GPP 與 DSP 間做資料傳遞的工作。
  - **PROCESSOR MANAGER :**

    PROCESSOR MANAGER 為 API 層，將 LINK DRIVER 的控制拉出並轉換提供給使用者使用。
  - **DSP/BIOS LINK API :**

    雖說 DSP/BIOS LINK API 這層是使用者在 GPP 端的使用介面，但大部分的 API 轉換其實都在 PROCESSOR MANAGER 這層

處理掉了，DSP/BIOS LINK API 這層只對呼叫時的輸入參數作確認的工作。

- **DSP 端 :**

DSP 端架構中的 DSP/BIOS 可視為在 DSP 上的小型 OS，而 LINK DRIVER 則是 DSP/BIOS 的許多 DRIVER 中的一個，負責處理跟 GPP 的溝通，DSP 端不需要額外的 API 層，因為溝通都是使用 DSP/BIOS 內有的模組(SIO/GIO/MSGQ)。

DSP/BIOS LINK 提供了基本的處理器控制、大量資料傳遞與短訊息傳遞功能，其 API 可分 PROC、CHNL、MSGQ 與 POOL 這四種類型，接下來分別介紹。

- **PROC :**
  - 初始化 DSP 處理器，使 DSP 到達 GPP 可以溝通的狀態。
  - 將程式載到 DSP 上。
  - 啟動與中斷 DSP 的執行。
  - 對 DSP 記憶體作讀取與寫入。
- **CHNL :**
  - 建立邏輯通道(logical channel)，此通道對應到 GPP 與 DSP 之間的物理連接(physical connectivity)，在 GPP 與 DSP 間做資料傳遞。
  - 使用要求-回應機制(issue-reclaim model)傳接資料。
- **MSGQ :**
  - 用來在 GPP 與 DSP 間交換少量資料，約為變數大小的短訊息。
  - 訊息的送出與接收都透過 message queues。
  - 允許單一發送端，但有多接收端。
- **POOL :**
  - 從共享記憶體(shared memory)中劃一塊記憶體池(memory pool)，以提供 CHNL 與 MSGQ 在此區塊內宣告暫存器。

3.4 軟體系統架構

雙核心系統分別是 ARM 和 DSP，由 ARM 負責主控系統運作、對周邊控制，DSP 負責運算相關演算法，但為了降低 DSP 晶片負擔，決定將人臉追蹤演算法移至 ARM 端實現，至於 ARM 與 DSP 雙核心之間的溝通方式，我們使用 DSP/BIOS LINK 來達成。

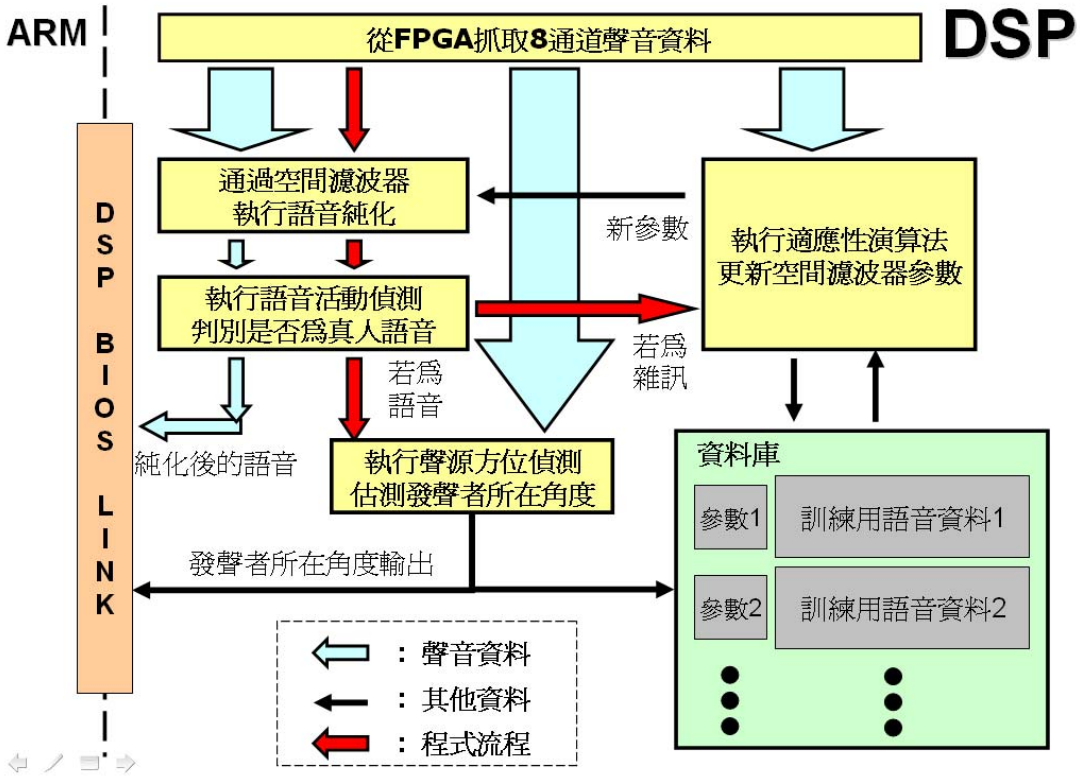


圖 3-17　DSP 端軟體系統流程圖

圖 3-17 為 DSP 端軟體系統流程圖，由於資料相關性單純、執行緒中不會有不定時間的等待現象、除了與 FPGA 外沒有跟其他周邊溝通，因此我們簡單使用單一執行緒（Single Task）去架構 DSP 端系統，當 FPGA 填滿一個內部暫存記憶體時，DSP 會透過外部記憶體介面直接向 FPGA 抓取八通道的聲音資料，取得聲音資料後直接通過 10 階的空間濾波器對聲音做純化的動作，經過空間濾波器後，聲音資料會從八通道合成單一通道聲音，並給語音活動偵測系統來判別是否為真人語音，由於純化後的聲音由

於大幅壓抑了背景雜訊能量，因此可提高語音活動偵測判斷正確率，而純化後的聲音資訊除了做語音活動偵測外，也會透過 DSP/BIOS LINK 的 Share Memory 機制送至 ARM 端供周邊使用，做即時性的輸出。

語音活動偵測的判斷結果，會引發兩種不同的程式流程，若語音活動偵測系統判斷該聲音不為語音而是背景雜訊，則空間濾波器參數更新系統會使用此聲音資訊對濾波器參數作訓練與更新，如此才能對不同環境不同時間點的背景雜訊做適應性的調整，空間濾波器參數更新系統會從資料庫中選一筆訓練用語音資料與參數出來，然後跑適應性演算法 NLMS 對參數作訓練，至於選取哪一筆訓練用語音資料與參數則是決定在於要對哪個方位做語音純化動作，也就是需要知道發聲者方位，而初始設定為陣列零度位置。

若語音活動偵測系統判斷該聲音為語音，接著我們會執行聲源方位偵測系統去估測發聲者方位，估測出的方位角度一來會提供給空間濾波器參數更新系統，作為選取哪一筆訓練用語音資料與參數的依據，二來也會傳至 ARM 端供系統周邊使用。完成整個程式流程後，DSP 會等待下一次抓取 FPGA 資料。

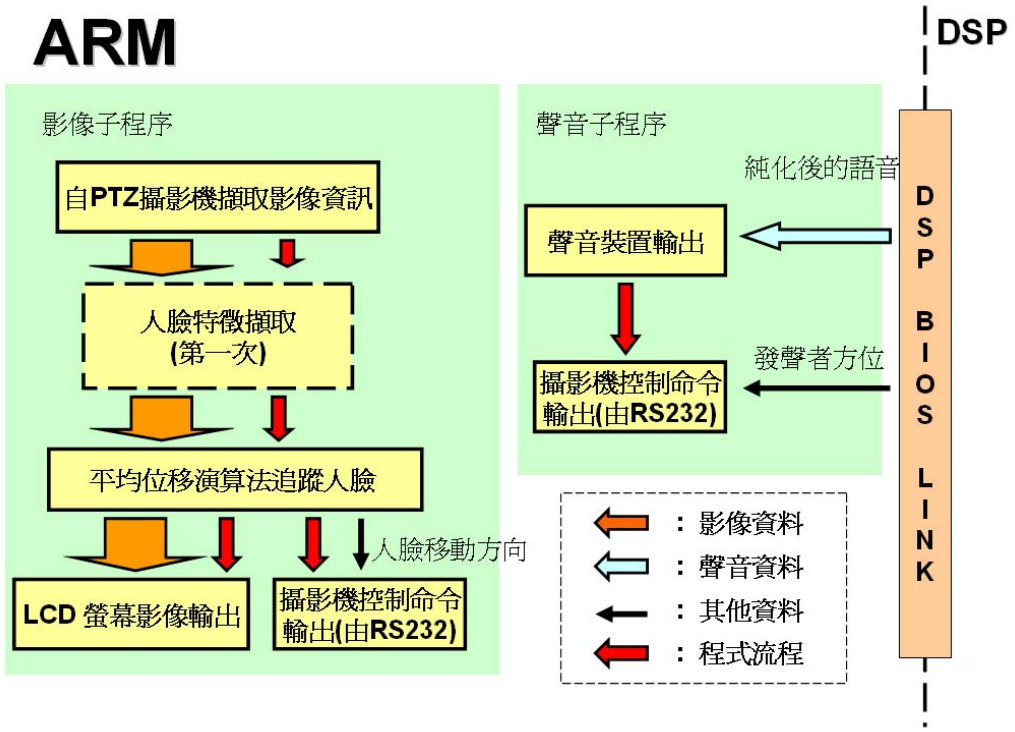


圖 3-18　ARM 端軟體系統流程圖

圖 3-18 為 ARM 端軟體系統流程圖，ARM 主要負責周邊控制，所以有一部分的程式是在跟周邊做溝通，我們把影像與聲音部分分割成兩個子程序執行，但聲音程序部分大多都在 DSP 實現了，因此在 ARM 只負責接收 DSP/BIOS LINK 傳遞的聲音資料並由 AUDIO OUT 做類比輸出，此外 ARM 在聲音子程序部分還會接收 DSP 傳來的發聲者方位資訊，透過 RS232 對 PTZ 攝影機做伺服控制，因此在聲音子程序中我們使用了兩個 pthread 去建構。

影像子程序先會接收 PTZ 攝影機的影像資訊，接著執行人臉追蹤，系統會在人臉第一次進入畫面時擷取人臉特徵，建立目標模型，爾後除非影像脫鎖，否則目標模型不再重建。平均位移演算法會不斷找出人臉移動方位，因此我們可以使用此資訊透過 RS232 對 PTZ 攝影機做伺服控制，使人臉保持在影像畫面正中央，達到追蹤效果。追蹤的影像畫面則透過 LCD 螢幕做即時顯示。在影像子程序部分，除了影像的輸入輸出外，我們額外開了一個 pthread 執行人臉追蹤演算法，一個 pthread 做攝影機伺服控制。

3.5 多通道聲音資料擷取

3.5.1 FPGA 端

數位式麥克風訊號透過 FPGA 平台處理後，再傳送至 DM6446 平台供後端演算法應用。FPGA 部分主要對輸入進來的 1-bit 資料作 Decimation 處理，之後存進各通道對應之 ping-pong buffer，並由 ping-pong control 加以控制。每當 ping-pong buffer 填滿時，FPGA 會送出 full 訊號通知 DM6446，DM6446 則開始送出 rdaddr(read address)，FPGA 端根據此 rdaddr 將對應的資料送出。其中 rdaddr 包含 sel 資訊以供多工器選擇通道，rdaddr 可直接視為一塊包含 8 個通道資料的記憶體。

FPGA 主要模組包含 Decimation、Ping-pong buffer、Ping-pong Control、Decode & Multiplexer，此外另有 Clock Divider 與 debounce 模組，以下將一一介紹其功能。圖 3-19 為 FPGA 主要架構方塊圖。

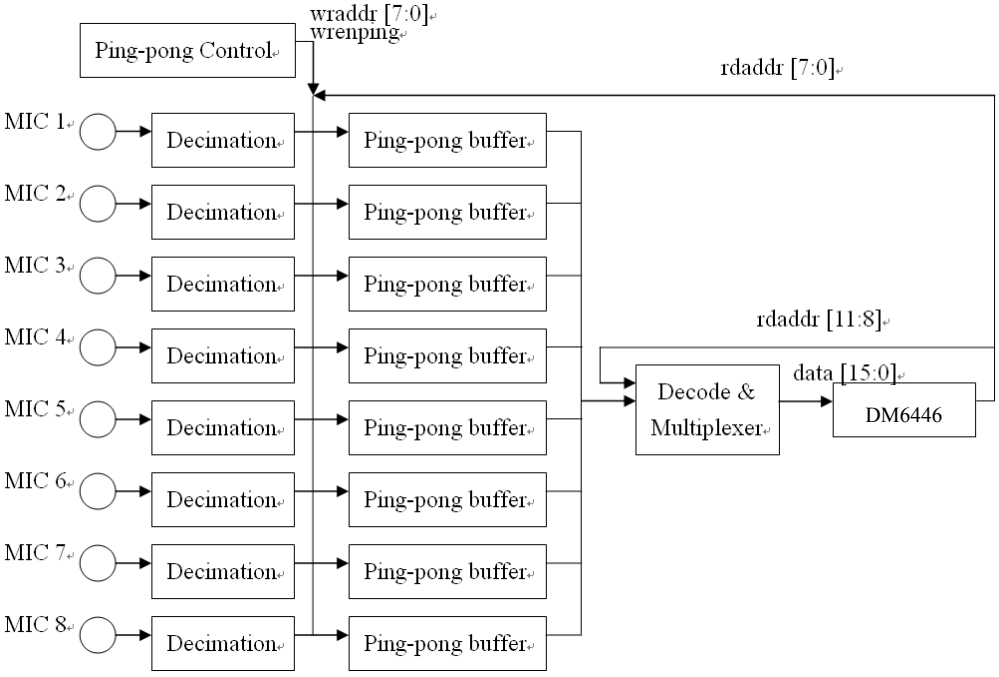


圖 3-19　FPGA 主要架構方塊圖

1. debounce：此模組只針對 reset 訊號做，目的是確保 reset 訊號不會被雜訊干擾而驅動。

2. Clock Divider：對外部震盪器時脈做除頻動作，一共提供 20 MHz、1.2 MHz 與 16 kHz 三種時脈。其中 20 MHz 用來做 rdclk(read clock)，1.2 MHz 提供給數位麥克風，16 kHz 則是最後訊號產生的時脈。
3. Decimation：主要分成低通濾波器(LPF)與降頻(Downsample)兩個部份。由於數位麥克風透過 Sigma-Delta 調變，將大部分雜訊能量分散至高頻頻帶，透過低通濾波器與降頻後，可移除大部分高頻之雜訊，並避免訊號混疊(aliasing)，這也是數位式麥克風抗雜訊的優點之一。在低通濾波器部份以 IIR 濾波器做設計，取代了原本 FIR 濾波器的設計，優點是節省暫存器個數，如圖 4-3 。由於 IIR 濾波器拆成積分與微分兩部份，為了使積分部份穩定，我們將其 pole 稍微移進單位圓。如此一來，亦可對訊號低頻處(70 Hz 以下)達到高通濾波器的作用而去除低頻雜訊。

   降頻部份則是由 1.2 MHz 降為 16 kHz，固降頻 75 倍。

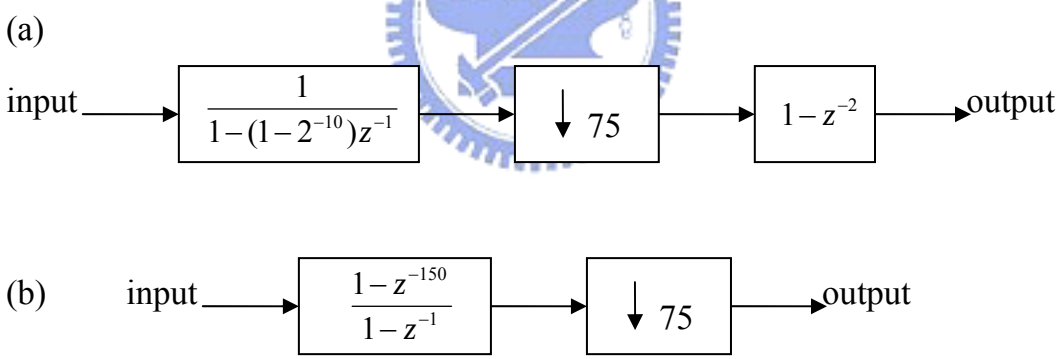


圖 3-20　Decimation 方塊圖 (a)IIR (b)FIR

4. Ping-pong buffer：Buffersize 為 256 (Int16)，透過 wrenping 判斷資料該讀寫在 ping buffer 或 pong buffer 中。若 wrenping 為 1，則表示資料可寫入 ping buffer 中，而 pong buffer 資料可讀取。資料寫入的時脈為 16 kHz，而讀取資料的時脈(rdclk)為 20 MHz。
5. Ping-pong Control：主要產生 wrenping、wraddr 與 full 訊號。wrenping 提供給各通道之 ping-pong buffer 來判斷讀寫的 buffer。wraddr 以簡單計數器產生，提供給各通道之 ping-pong buffer 寫入資料的位置。full

訊號則是在 ping-pong 切換時產生，以告知 DM6446 抓取資料。full 訊號並非完全與 ping-pong 切換訊號(wrenping)同步，有加入一點延遲，確保讀寫間的安全。

6. Decode & Multiplexer：判斷 DM6446 輸入的 rdaddr，並切換對應的通道資料。

3.5.2 DM6446 端

對 DM6446 平台而言，八個通道的聲音資料可視為是儲存在一塊連續的外部記憶體之中，而平台透過其非同步外部記憶體介面向此外部記憶體抓取聲音資料。由 FPGA 端架構一節可知 FPGA 模擬成一塊大小為 4096 bytes(8 channel x 256 buffer size x 2 bytes)的記憶體，但由於使用 ping-pong 機制，FPGA 實際上是使用兩倍大的記憶體空間(8192 bytes)。

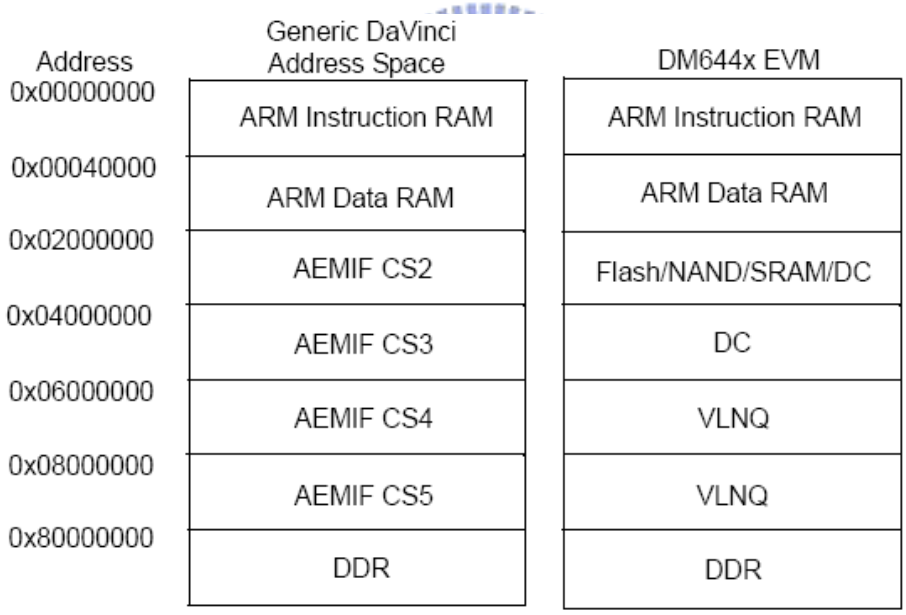


圖 3-21 DM6446 記憶體位置空間對應圖

上圖 3-21 為 DM6446 的記憶體位置空間對應圖[18]，由圖中可看出非同步外部記憶體介面(AEMIF)對應到的記憶體位置由 0x02000000 到 0x08000000， 其中 AEMIF CS3 (0x04000000 ~ 0x06000000)這區塊是對應到硬體的 DC(Daughter Card)，較為適合拉出來與 FPGA 連接，因此決定使用 AEMIF CS3 這塊記憶體位置空間去對應到 FPGA，而在實際使用上，先

是在 DSP 的記憶體配置檔案(.cmd)中由 AEMIF CS3 記憶體位置空間配置一塊大小為 4096 bytes 的暫存器，此記憶體位置區間需與 FPGA 端事先溝通好，如此才能確保 FPGA 端的資料對應無誤，之後流程則是 DM6446 判斷 FPGA 傳出的 full 訊號，一但 full 則由此暫存器搬資料到 DSP 內部暫存器即可。下圖 3-22 為 AEMIF DC 的腳位對應圖[18]，其中 B.EM_A 即為對應的 address bus 而 EM_D 為對應的 data bus。

Pin #	Signal	Pin #	Signal
1	B.EM_A21	2	B.EM_A20
3	B.EM_A19	4	B.EM_A18
5	B.EM_A17	6	B.EM_A16
7	B.EM_A15	8	B.EM_A14
9	GND	10	GND
11	B.EM_A13	12	B.EM_A12
13	B.EM_A11	14	B.EM_A10
15	B.EM_A9	16	B.EM_A8
17	B.EM_A7	18	B.EM_A6
19	GND	20	GND
21	B.EM_A5	22	B.EM_A4
23	B.EM_A3	24	CLE_EM_A4
25	B.EM_A1	26	ATA2_EM_A0
27	GND	28	GND
29	ATA_CS1	30	ATA_CS0
31	ATA1_EM_BA1	32	ATA0_EM_BA0
33	WRITE_WE	34	READ_OE
35	WAIT/BUSY	36	INTRQ_EM_RNW
37	GND	38	GND
39	EM_D15	40	EM_D14
41	EM_D13	42	EM_D12
43	EM_D11	44	EM_D10
45	GND	46	GND
47	EM_D9	48	EM_D8
49	EM_D7	50	EM_D6
51	EM_D5	52	EM_D4
53	GND	54	GND
55	EM_D3	56	EM_D2
57	EM_D1	58	EM_D0
59	EM_CS3	60	DC_EM_CS2
61	1.8V.SYS_RESETz	62	CLKOUT0
63	GND	64	GND
65	VCC_3.3V	66	VCC_3.3V
67	GND	68	GND
69	VCC_5V	70	VCC_1.8V

圖 3-22 AEMIF DC 腳位對應圖

AEMIF[19]的data bus寬度有8bits跟16bits兩種可選，由於FPGA資料格式是16bits(int16)，所以AEMIF對應使用16bits的data bus寬度。AEMIF的傳輸速度是可以控制的，但由於是非同步架構，所以速度的調整並非直接改變單一的時脈。每一個讀取或寫入的時序都可分為三個階段，分別是Setup階段、Strobe階段與Hold階段，每一階段都有一個小暫存器控制各階段時間長度，實際的時間長度為暫存器值乘上AEMIF的internal clock rate，但此internal clock rate是不可控的。由於FPGA是我們自行設計的硬體，因此DM6446與FPGA的通訊機制可以不完全照著AEMIF的機制做溝通，所以我們直接藉由分別調快Setup、Strobe、Hold三階段時間來加快AEMIF速度與FPGA溝通。我們修改Linux kernel 程式來加快AEMIF速度，在程式中加上對Asynchronous 3 Configuration Register此暫存器做設定的部份即可(Asynchronous 3 Configuration Register 對應到 AEMIF CS3)，32Bits暫存器中各Bit對應到的意義可由下表3-1得知。修改Bit 1-0可以調整data bus 寬度，而Bit 29- 4 可以加快AEMIF的讀寫速度。

Bit	Field	Value	Description
31	SS		Select Strobe bit. This bit defines whether the asynchronous interface operates in Normal mode or Select Strobe mode. See Section 2.5 for details on the two modes of operation.
		0	Normal mode is enabled.
		1	Select Strobe mode is enabled.
30	EW		Extend Wait bit. This bit defines whether extended wait cycles will be enabled. See Section 2.5.8 on extended wait cycles for details. This bit field must be cleared to 0, if the EMIF on your device does not have a EM_WAIT pin.
		0	Extended wait cycles are disabled.
		1	Extended wait cycles are enabled.
29-26	W_SETUP	0-Fh	Write setup width in EMIF clock cycles, minus 1 cycle. See Section 2.5.3 for details.
25-20	W_STROBE	0-3Fh	Write strobe width in EMIF clock cycles, minus 1 cycle. See Section 2.5.3 for details.
19-17	W_HOLD	0-7h	Write hold width in EMIF clock cycles, minus 1 cycle. See Section 2.5.3 for details.
16-13	R_SETUP	0-Fh	Read setup width in EMIF clock cycles, minus 1 cycle. See Section 2.5.3 for details.
12-7	R_STROBE	0-3Fh	Read strobe width in EMIF clock cycles, minus 1 cycle. See Section 2.5.3 for details.
6-4	R_HOLD	0-7h	Read hold width in EMIF clock cycles, minus 1 cycle. See Section 2.5.3 for details.
3-2	TA	0-3h	Minimum Turn-Around time. This field defines the minimum number of EMIF clock cycles between the end of one asynchronous access and the start of another, minus 1 cycle. This delay is not incurred by a read followed by a read or a write followed by a write to the same CS space. See Section 2.5.3 for details.
1-0	ASIZE	0-3h	Asynchronous data bus width. This bit defines the width of the asynchronous device's data bus.
		0	8-bit data bus
		1h	16-bit data bus
		2h-3h	Reserved

表 3-1　A3CR 暫存器設定表

第四章　系統測試與結論

4.1 語音活動偵測功能測試

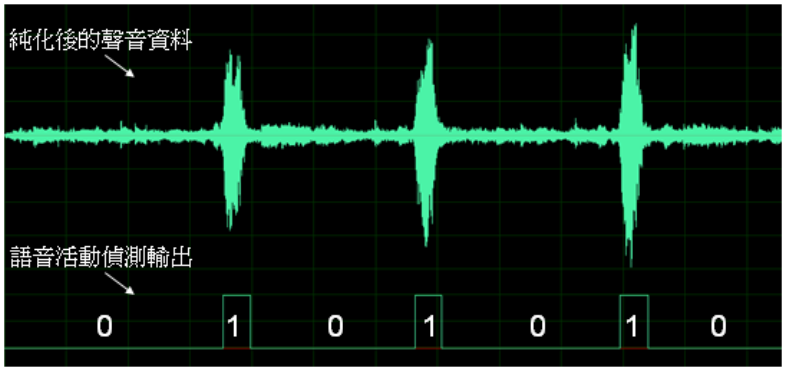


圖 4-1 語音活動偵測測試圖

圖 4-1 為語音活動偵測測試圖，測試內容為將經純化的聲音資訊送至平台的語音活動偵測系統判別是否為真人語音，圖中的語音活動偵測輸出即為判別結果，1 代表有真人語音，0 代表無真人語音，從圖中可以看出系統判別正確。

4.2 聲源方位估測系統測試

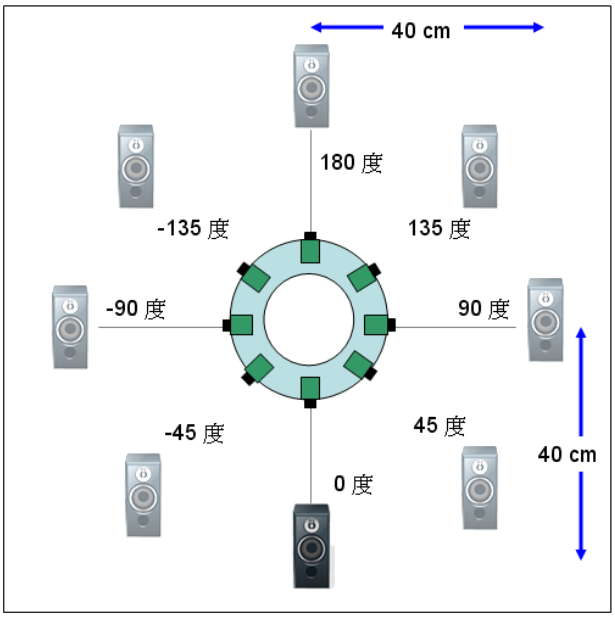


圖 4-2 聲源方位估測實驗環境示意圖

圖 4-2 為聲源方位估測實驗的實驗環境示意圖，我們在距離環型數位式麥克風陣列 40 公分處擺放喇叭作為發聲源，讓喇叭在同一方位連續播放 10 次的”喂”的預錄人聲，此時系統會判斷出 10 個角度，紀錄這 10 個角度作為這個方位的系統測試結果。實驗環繞著環型陣列由 0 度開始每隔 45 度重複上述步驟，量測正負 180 度，以下為實驗結果。

	0 度角量測	45 度角量測	90 度角量測	135 度角量測	180 度角量測	-135 度角量測	-90 度角量測	-45 度角量測
1	1	45	90	136	179	-135	-91	-45
2	2	45	90	137	178	-137	-88	-46
3	0	45	88	137	178	-136	-90	-45
4	-1	47	89	138	179	-136	-88	-44
5	1	45	90	138	180	-135	-95	-43
6	1	47	88	138	180	-135	-91	-48
7	2	47	88	137	180	-136	-95	-47
8	-1	46	90	137	178	-135	-91	-47
9	2	45	88	137	177	-135	-93	-44
10	0	44	90	135	180	-136	-90	-45
平均角度	0.7	45.6	89.1	137	178.9	-135.6	-91.2	-45.4

表 4-1 聲源方位估測實驗結果

由實驗結果可以看出，各方位估測出的角度中，除了-90 度的方位有兩次估測成 95 度外，其餘的誤差都在正負三度內。而其中平均以-45 度方位的估測結果最佳，只有-0.4 度的誤差。有許多因素都會導致影響角度估測結果，如，喇叭不是完美的點聲源、聲速於不同溫度會改變，與我們實驗計算時帶入的 340m/sec 會有些許差異，都會導致估測結果有誤差，但在系統應用上，人與系統不會相距太遠，角度的誤差不會因為距離而被放大太多，因此正負三度的誤差是在可以接受的範圍。

4.3 語音純化系統測試

4.3.1 空間濾波器階數對純化效果影響測試

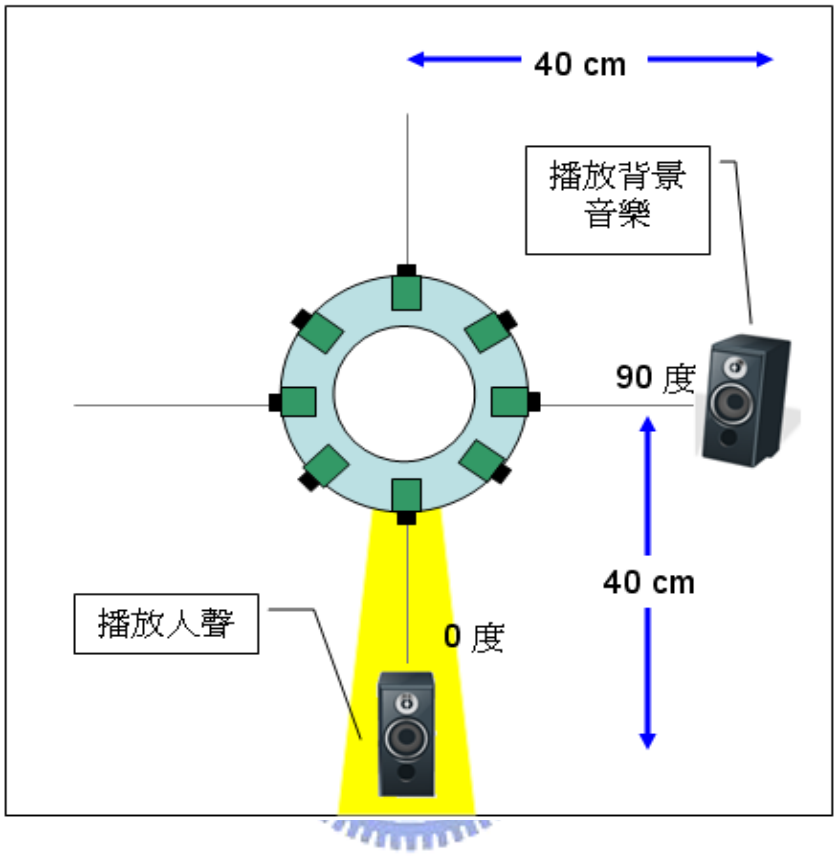


圖 4-3 空間濾波器階數對純化效果影響測試實驗環境示意圖

圖 4-3 為空間濾波器階數對純化效果影響測試的實驗環境示意圖，我們在距離環型數位式麥克風陣列 40 公分處的 0 度位置與 90 度位置各擺放一個喇叭作為發聲源，90 度位置喇叭持續播放著流行音樂作為背景雜訊，0 度位置喇叭則播放著語音，背景音樂與語音能量皆固定，且讓純化系統會對 0 度位置作語音作增強的動作，測試不同的空間濾波器階數對語音純化效果的影響。表 4-2 為分別對 1 階、5 階、8 階、10 階、12 階、15 階與 30 階濾波器階數作語音純化實驗的結果，每一個階數皆作五次的提升訊噪比量測並取其平均。圖 4-4 為表 4-2 中各階濾波器提升的訊噪比的直條圖。

階數		平均人聲能量 (dB)	平均背景音樂能量 (dB)	平均 訊噪比	平均 提升的訊噪比
1	純化前	-17.624	-23.124	5.5	8.036
	純化後	-17.574	-31.11	13.536	
5	純化前	-17.802	-22.732	4.93	10.222
	純化後	-18.436	-33.588	15.152	
8	純化前	-17.76	-22.868	5.108	10.934
	純化後	-18.436	-34.478	16.042	
10	純化前	-17.722	-22.738	5.016	11.286
	純化後	-18.51	-34.812	16.302	
12	純化前	-17.724	-22.688	4.964	11.15
	純化後	-18.558	-34.672	16.114	
15	純化前	-17.804	-23.06	5.256	10.836
	純化後	-19.014	-35.106	16.092	
30	純化前	-17.748	-22.81	5.062	10.758
	純化後	-19.922	-35.742	15.82	

表 4-2 空間濾波器階數對純化效果影響

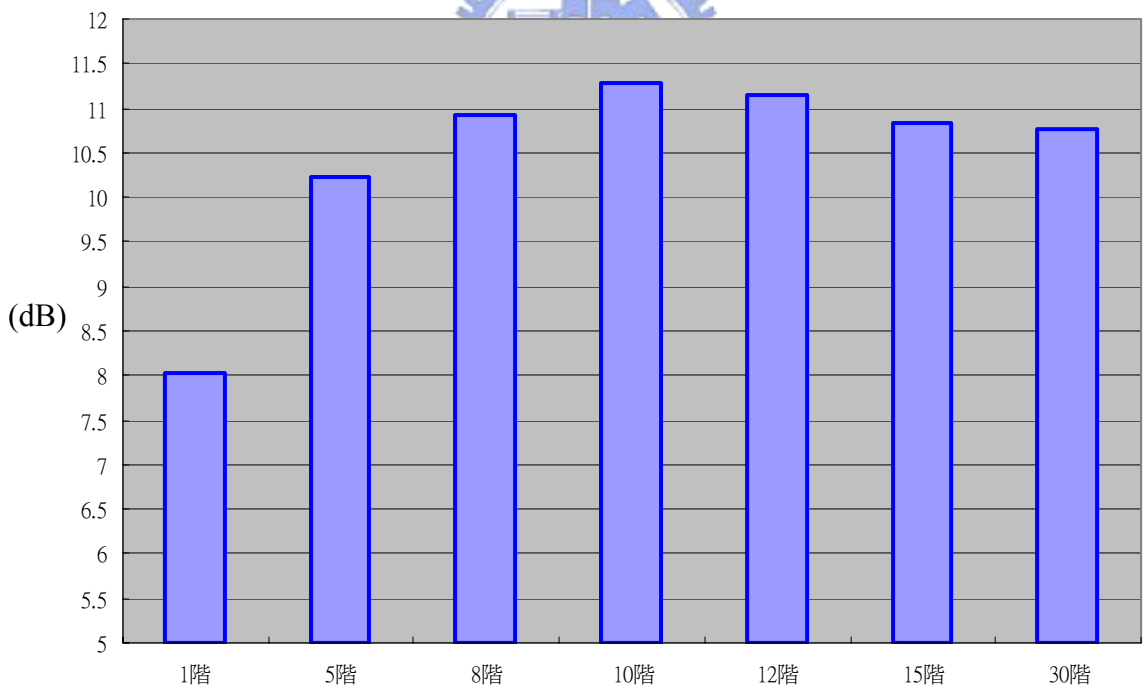


圖 4-4 各階數提升訊噪比直條圖

觀察實驗結果可以發現，10 階的空間濾波器會有最佳的訊噪比提升，當濾波器階數小於 10 階時，階數越小則抑制的背景音樂(雜訊)的效果越差，理由是階數不足無法完整補償空間特性與麥克風特性，抑制效果越

弱，反觀，若濾波器階數大於 10 階時，會有不錯的抑制背景音樂(雜訊)的能力，但由於階數過大，產生 Over-fitting 現象，所以也較為抑制語音能量，因此相較之下，訊噪比提升量會較 10 階濾波器來得差，所以 10 階的濾波器能大量抑制背景音樂，且不會過度抑制語音，為實驗結果中最佳的濾波器階數。

4.3.2 背景音樂能量對純化效果影響測試

背景音樂能量對純化效果影響測試的實驗環境與 4.3.1 節的實驗環境相同，只是實驗中固定濾波器階數為 10 階，改變背景音樂能量，同一種背景音樂能量做五次純化實驗，取其平均。下表 4-3 為實驗結果。

實驗 No		平均人聲能量 (dB)	平均背景音樂能量 (dB)	平均訊噪比	平均提升的訊噪比
1	純化前	-19.152	***-42.098***	22.946	4.178
	純化後	-18.328	-45.452	27.124	
2	純化前	-19.044	***-32.626***	13.582	9.352
	純化後	-18.288	-41.222	22.934	
3	純化前	-18.558	***-26.62***	8.062	11.442
	純化後	-18.336	-37.84	19.504	
4	純化前	-18.042	***-23.532***	5.49	11.888
	純化後	-18.526	-35.904	17.378	
5	純化前	-17.126	***-20.642***	3.516	11.77
	純化後	-18.624	-33.91	15.286	
6	純化前	-16.518	***-19.458***	2.94	11.048
	純化後	-18.652	-32.64	13.988	

表 4-3 背景音樂能量對純化效果影響

由表中可以看出，最大的提升訊噪比是發生在背景音樂能量為 -23.532dB 時，此時提升的訊噪比為 11.888dB，若背景音樂能量太小，直觀上可知，因可以壓抑的能量有限，所以提昇訊噪比不多，但由實驗得知，若背景音樂能量太大，系統試圖壓抑背景音樂的同時也會壓抑掉語音訊號，所以平均提升的訊噪比也會較差。下圖 4-5 為實驗過程中為最佳的純化結果的音檔，上半圖是純化前，下半圖是純化後結果，經純化處理後訊噪比提升了 12.09dB。

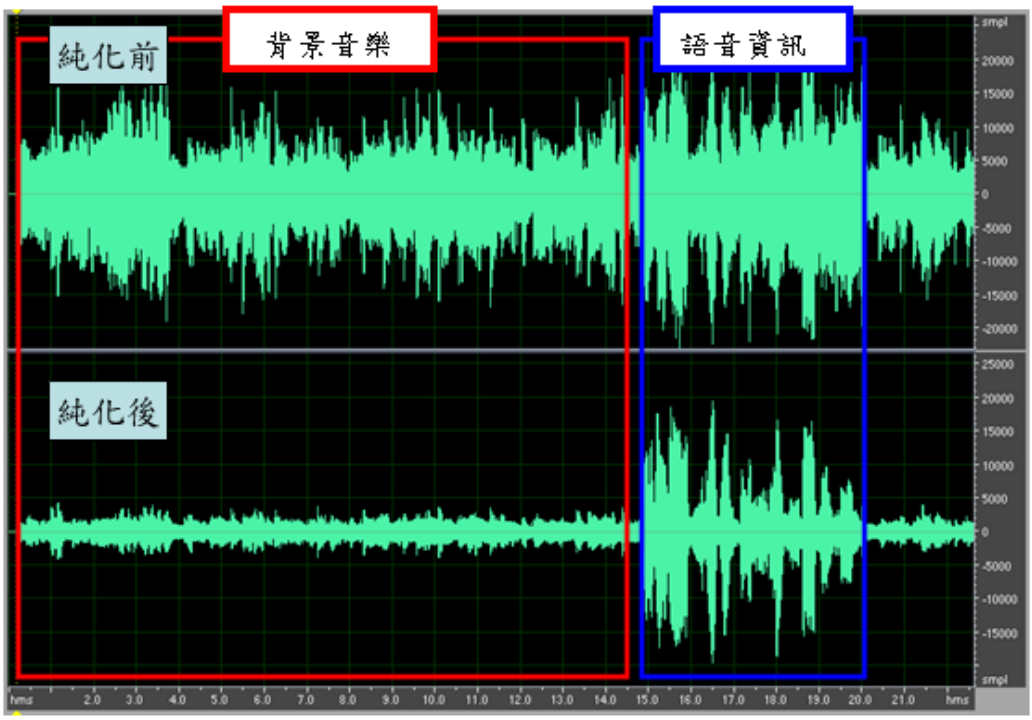


圖 4-5 純化前後比較圖(提升 12.09dB)

4.3.3 背景音樂位置對純化效果影響測試

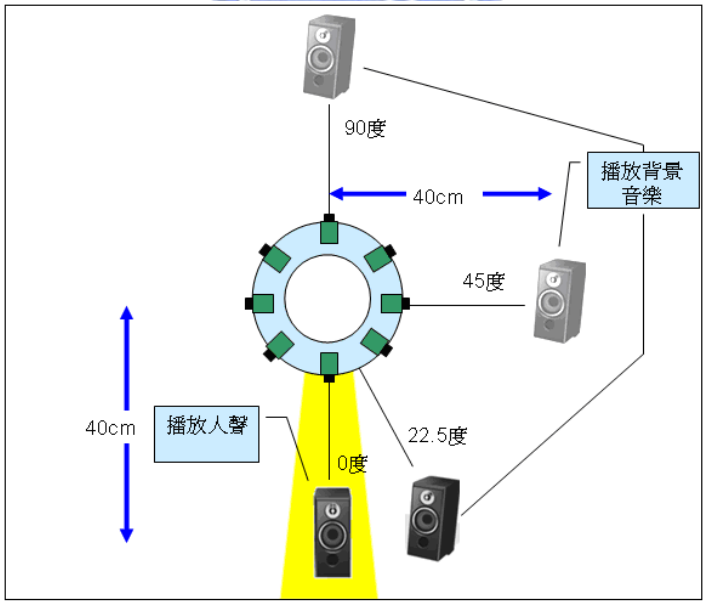


圖 4-6 背景音樂位置對純化效果影響實驗環境示意圖

圖 4-6 為背景音樂位置對純化效果影響測試實驗環境示意圖，我們在距離環型數位式麥克風陣列 40 公分處的 0 度位置擺放一個喇叭作為發聲源，播放著語音，把播放背景音樂的喇叭分別擺在 22.5 度、90 度、180 度

位置，純化系統對 0 度位置作語音作增強的動作，而背景音樂、語音能量、濾波器階數皆固定，表 4-4 為實驗結果。

背景音樂位置		平均人聲能量 (dB)	平均背景音樂能量 (dB)	平均訊噪比	平均提升的訊噪比
180 度	純化前	-17.52	-22.468	4.948	11.09
	純化後	-18.61	-34.648	16.038	
90 度	純化前	-16.914	-21.052	4.138	10.658
	純化後	-18.37	-33.166	14.796	
22.5 度	純化前	-16.438	-18.802	2.364	6.82
	純化後	-21.528	-30.712	9.184	

表 4-4 背景音樂位置對純化效果影響

由實驗結果可以看出背景音樂與語音所在位置夾角越大效果越好，這是合理的實驗結果，因為系統會壓抑背景音樂方向的聲音，所以若語音方向與背景音樂方向夾角越小，語音越容易也被系統壓抑掉，所以純化效果下降。

4.4 人臉追蹤系統測試

圖 4-7 系統人臉追蹤測試

圖 4-7 為追蹤系統測試追蹤人臉的結果，每張圖間隔時間為一秒，順序為由上而下由左而右，綠色框框即為系統判別之人臉位置，由測試結果可以看出人臉追蹤系統可以持續並正確的框出人臉所在位置。由於人臉追蹤系統也可以追蹤其他的物體，因此我們讓系統分別追蹤杯子與娃娃來加以測試，發現系統的確是可以持續並正確框出物體所在位置，圖 4-8 與圖 4-9 為測試結果圖，一樣每張是圖間隔時間為一秒，順序為由上而下由左而右，綠色框框即為系統判別之物體位置。

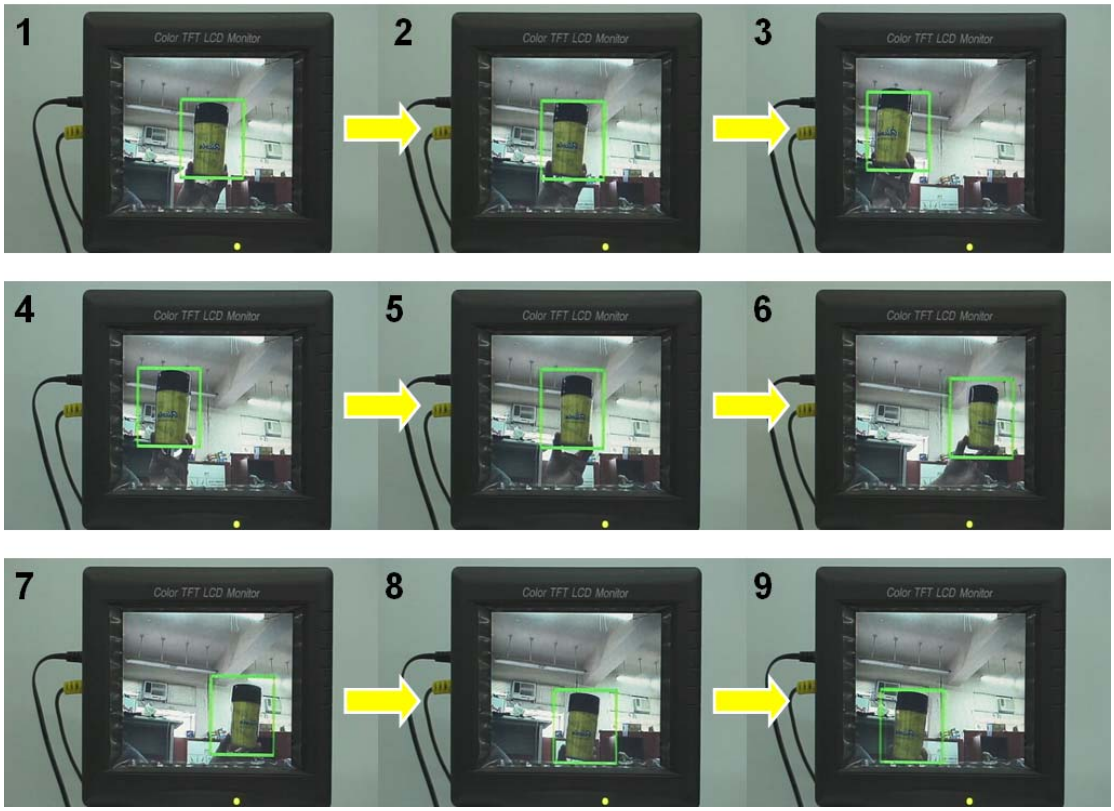


圖 4-8 系統物體追蹤測試_杯子

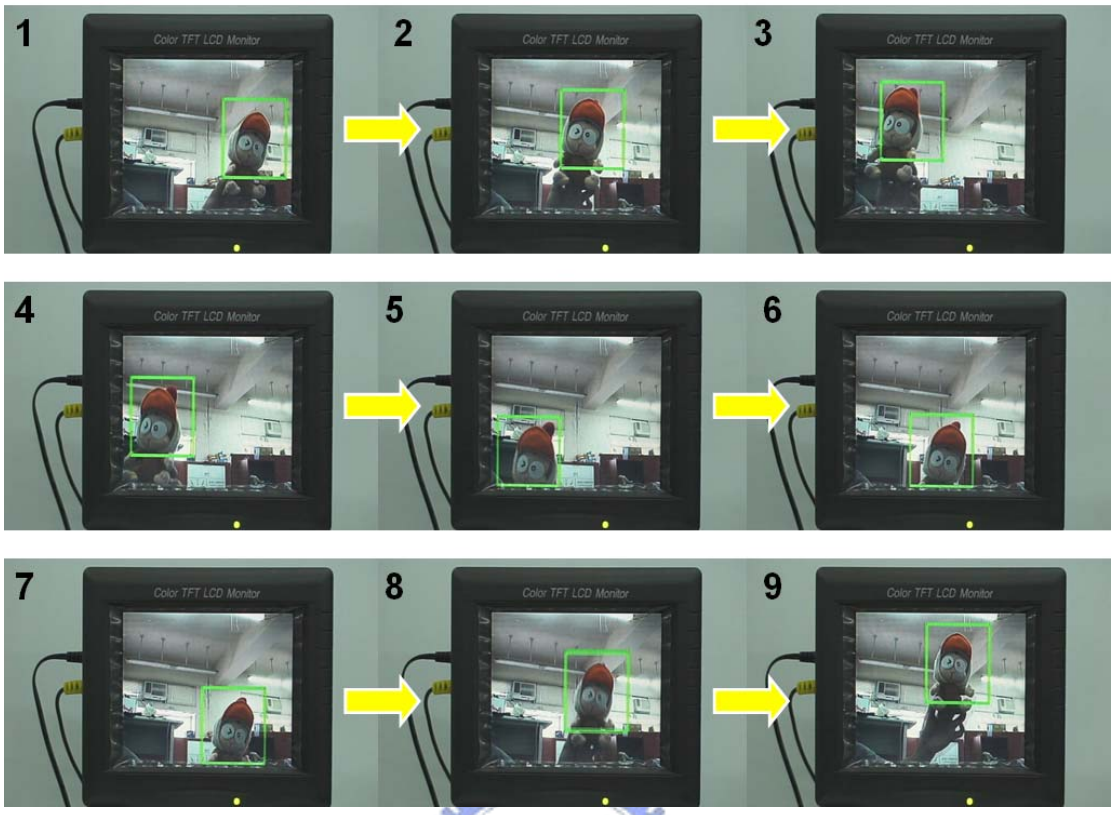


圖 4-9 系統物體追蹤測試_娃娃

4.5 結論與未來展望

本論文已實作完成在 DM6446 平台上實現影音追蹤與語音純化系統，聲源方位估測系統估測出的聲源角度之誤差能低於正負三度，此精準度足以幫助影像系統解決影像脫鎖之問題，達到以聲音系統輔助影像系統的功效，語音純化系統在背景噪音強的情況下有 11~12dB 的訊噪比提升，可以大幅抑制雜訊，強化語者語音資訊，提升語音品質，而人臉追蹤系統可即時且持續性的追蹤人臉特徵。透過在平台上實現這三大功能，本論文建立起一套高整合性且多應用面之影音人機互動裝置。

本系統仍有許多可以在發展的空間，目前聲源方位估測系統是估測水平方位，而系統可以透過演算法的修改加強，進而估測三維立體空間方位。目前系統是透過聲音輔助影像，未來也可以加上影像輔助聲音功能，以提高發聲者所在位置估測的精準度，透過上述開發，皆可將系統的智慧型功能再往上提升一個層次。

參考文獻

[1] Javier Ramírez , José C. Segura , Carmen Benítez , Ángel de la Torre and Antonio Rubio ,"Efficient voice activity detection algorithms using long-term speech information," Speech Communication, Volume 42, Issues 3-4, April 2004, Pages271-287.

[2] European Digital Cellular Telecommuni- cations System; Half rate speech; Voice Activity Detection (VAD), ETSI GSM 06.42 (ETS 300-581-6), 1995.

[3] European Digital Cellular Telecommuni- cations System; Half rate speech; Half rate speech transcoding, ETSI GSM 06.20 (ETS 300-581-2), 1995.

[4] ITU-T G.729, Coding of Speech at 8kbit/s Using CS-ACELP, March, 1996.

[5] A. Benyassine, E. Shlomot, H. Y. Su, D. Massaloux, C. Lamblin, and J. P. Petit, "ITU recommendation G.729 annex B: A silence compression scheme for use with G.729 optimized for V.70 digital simultaneous voice and data applications," IEEE Commun. Mag., vol. 35, pp. 64–73, Sept. 1997.

[6] 林家銘, *以數位訊號處理器為架構隻智慧型天線接收機雛形實現*, 交大碩士論文,2002.

[7] 鍾青衛, *智慧型天線接收機之 DOA 與 Beamforming 演算法即時實現*, 交大碩士文,2003.

[8] S.U. Pillai, *Array Signal Processing*, Springer-Verlag, 1989.

[9] Ta-Sung Lee, *Array Signal Processing*, course sheet, 2002.

[10] Petre Stoica and Torsten Soderstrom, Statistical analysis of MUSIC and ESPRIT estimates of sinusoidal frequencies, *International Conference, Acoustics, Speech, and Signal Processi* 14-17 April 1991.

[11] 楊佳興, *使用麥克風陣列實現即時語音純化與真人語音活動偵測系統*, 交大碩士論文,2005.

[12] 梁開泰, "智慧型天線及其應用", *通訊雜誌*, 第 69 期, 10 月號, 1999.

[13] 張家瑋, *Multi-DSP 平台應用於 DOA 與 Beamforming 之即時模擬系統*, 交大碩士論文,2004.

[14] 阮崇維,*使用空間與顏色特徵的平均移動演算法於物件大小與方位追蹤*, 交大碩士論文,2007.

[15] TI, DVEVMGetting Started Guide ,Literature Number:SPRUE66A , August 2006.

[16] TI, DM644x DMSoC ARM Subsystem Reference Guide ,Literature Number:SPRUE14 , 2005.

[17] TI, DSP/BIOS™ LINK USER'S GUIDE ,Version 1.30.06 ,NOV 22, 2005.

[18] TI, DaVinciEVM_TechRef, October 2006.

[19] TI, *TMS320DM644x DMSoC Asynchronous External Memory Interface (EMIF) User's* Guide, Literature Number: SPRUE20A June 2006.

[20] TI, DaVinciEVM Schematic.

[21] 孫蘭蕙,*以OMAP5912實現語者方位判定於輪式機器人平台*, 交大碩士論文,2006.