使用 ANN 抖音參數模型之國語歌聲合成

古鴻炎、林正甫

國立台灣科技大學資訊工程系
E-mail: guhy@mail.ntust.edu.tw

摘要

本文對於歌聲表情的一個重要因素“抖音”，研究以短時傅利葉轉換和解析信號之方法來對歌聲音節作分析，而求得抖音參數。求得訓練用歌曲的所有音節的抖音參數之後，再拿去訓練各項抖音參數分別的類神經網路(ANN)模型。之後依據所建造的 ANN 模型的輸出，再配合其它樂譜資料(速度、拍數)，去控制諧波加雜音信號模型(HNM)作歌聲信號的合成。經由主觀的自然度聽測實驗，所得的評分顯示，使用抖音參數合成出的歌聲信號，的確可以比原始使用 HNM 合成出的歌聲信號有顯著的改進。

1. 導言

過去已有一些歌聲信號合成的技術被提出，這些技術包括了 Phase Vocoder [1, 2]、Formant Synthesizer [1, 2]、ABS/OLA Sinusoidal Model [3]、PSOLA Synthesis [4]、EpR Model [5]。此外，我們也研究了一種以 HNM (harmonic-plus-noise model)爲基礎並加以改進的國語歌聲合成方法[6]。以今日的技術水準來說，要合成出乾淨無雜音、音質清晰的歌唱聲信號，已經不是困難的事了，但是電腦所合成出的歌聲，聽起來的感覺，並不像真人歌手所唱的那麼地具有歌聲表情(expression)的呈現，事實上電腦合成的歌聲，經常令人有機械腔調的感覺。造成這種情況的主要原因是，用以表達歌聲表情的一些因素未被適當的塑模(modeling)及控制，和歌聲表情之表達有關的因素包括: 抖音 (vibrato)、強調突顯 (marcato)、壓抑減弱 (soffocato)、彈性速度 (rubato)、漸快速度 (accelerando)、漸慢速度(ritardando)、…等等，其中”抖音”是一個很重要的因素，因此我們決定去分析歌聲裡抖音的參數數值，然後據以建立歌聲的抖音模型，希望使用此模型產生出的抖音參數，能夠合成出具有自然的抖音呈現的國語歌聲。

依據 Horii[7]和 Imaizumi[8]等人的研究，由”抖音”所引起的最顯著的聲學(acoustic)現象是，音高(pitch)頻率值會呈現近似週期性的擺動，一個例子如圖 1 裡所畫出的實線曲線，是由分析真人所唱的一個音節而得到，它的頻率值會隨著時間(橫軸)在 265Hz 至 285Hz 之間上下擺動，且擺動率約爲 5Hz。所以，要合成出具有抖音表情的歌聲信號，音高頻率是一個主要的需加以控制的聲學因素。另外，由圖 1 也可觀察到一個現象，即瞬間頻率高時，信號振幅也較大。

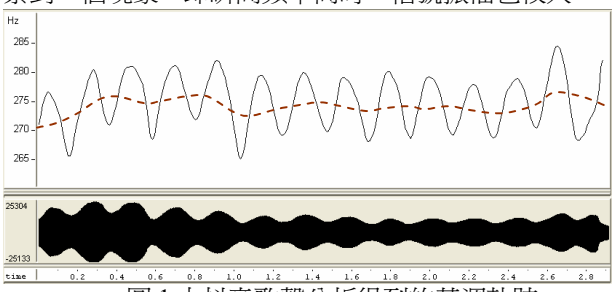

圖 1 由抖音歌聲分析得到的基週軌跡

雖然一條具有抖動特性的基週軌跡，可以應用一些規則來產生[1, 17, 18]，但是如此得到的基週軌跡，其抖音方式是否可以像真人所唱的那樣自然，這是令人懷疑的。因此，我們決定採用 ANN (artificial neural network)來建造歌聲抖音的模型，此模型並不是要直接用來產生具有抖動特性的基週軌跡，而是用來產生它所對應的抖音參數數值，然後利用這些參數數值去間接地產生出基週軌跡。這樣的作法是可實行的，因爲根據 Sundberg[9] 、Shonle 和 Horan[10]等人的研究論文，一條抖動的基週軌跡，可以被分析及使用三種參數來加以描述，這三種參數分別是音位軌跡(intonation)、抖音範圍(vibrato extent)、和抖音頻率(vibrato rate)。音位軌跡就是隨著時間變化較緩慢的平均音高曲線，如圖 1 裡所畫的虛線曲線；抖音範圍就是實線曲線的峰值減掉虛線高度後的差值；而抖音頻率是基週軌跡的擺動率。

當藉由 ANN 抖音參數模型輸出的數值去產生出一條基週軌跡之後，接著就可依據這條基週軌跡去計算音高調整過的 HNM 模型的參數數值 [6]，然後具有抖音表情的歌聲，就可以使用我們先前研究、改進的 HNM 合成方法[6]，去合成出信號波形。HNM 信號模型原先是由 Y. Stylianou 所提出來 [11, 12]，HNM 模型

對於位於高頻帶的雜音信號成分，可說是比弦波模型具有較好的 modeling 能力。

2. 抖音參數分析

在訓練 ANN 抖音參數模型之前，我們必需先收集真人歌手所唱的歌聲信號，並且分析出抖音參數數值。實際上在訓練階段裡，我們是依據如圖 2 所示的流程，去作抖音參數的分析處理，然後才去作 ANN 模型的訓練。首先，我們邀請一位真人歌手來唱歌並且錄音；然後，對所錄的歌聲裡的各個音節，以手工方式作音節界限和發音音標的標記，再依界限標記把各個音節的信號分割成各自的音檔；接著對各個音節的音檔，去量測它的瞬間音高頻率(instantaneous pitch frequency, IPF)的曲線；依據 IPF 曲線，再去分析出音位軌跡、抖音範圍、和抖音頻率等參數。前述的幾個處理步驟，其細節將會在以下的幾個子節中作說明，而 ANN 模型的訓練，則留在第 3 節中說明。

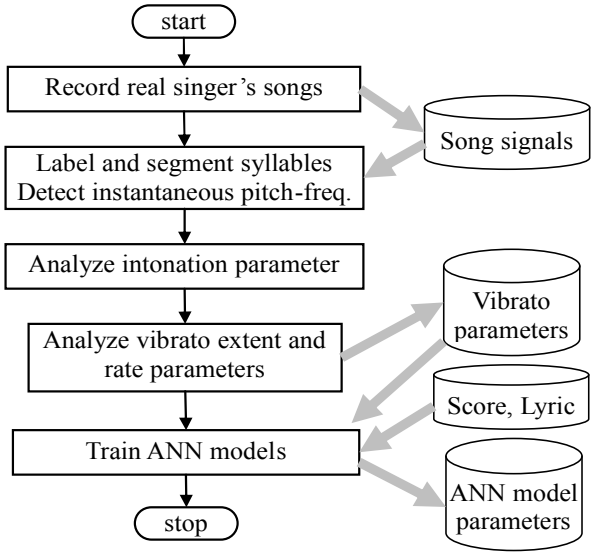

圖 2 訓練階段的處理流程

2.1. 歌聲信號錄音

在本研究裡，我們邀請了一位男性歌者，到一座 Acoustic-Syatem RE-242 隔音室，演唱流行歌曲，同一時間直接把歌聲信號錄存到電腦裡，取樣率是 22,050Hz。因爲他戴著耳機，且隨著播放給他聽的 MIDI 伴奏來唱歌，所以各個歌詞所唱出的音高基本上是正確的。我們分次收集了他演唱的 15 首歌曲，這些歌曲中有一些是快節奏的，也有一些是慢節奏的，音節的數量方面，則共有 2,841 個歌聲音節。

2.2. 瞬間音高頻率之量測

錄到真人演唱的歌聲之後，我們使用 WaveSurfer 軟體來對音節的界限和發音音標作手工標記的工作，標記完後，再把各個音節的信號分割成各自的音檔。由於一個國語音節起始部分的聲母，可能是個無聲(unvoiced)子音，因此我們必需先偵測出音節內無聲、有聲(voiced)兩部分的邊界點，然後才去對音節的有聲部分去量測瞬間音高軌跡 IPF 曲線。

IPF 量測的方法如下: (a)首先把音節有聲部分，切割成一序列有重疊的音框，音框的長度設爲 512 點，而相鄰音框每次只前進 32 點；(b)對於各個音框，套上 Hamming 窗後，於後面補零使成爲 4,096 點，再作 4,096 點的快速傅利葉(FFT) 計算；(c)從 FFT 頻譜上，尋找前五個頻譜峰點(spectral peak)的頻率值，第 i 個的值就除以 i 以作爲基頻的估計值，然後取這 5 個值的幾何平均值，作爲此音框的 IPF 值。

當全部音框的 IPF 值都求出後，將它們依音框次序串連起來，就可以得到 IPF 曲線 $f(t)$，$f(t)$可以看成是具有如下的型式 [9, 10]，

$$f(t) = V_d(t) + V_e(t) \cdot cos(\phi(t)) \tag{1}$$

其中 $V_d(t)$表示 $f(t)$的音位軌跡參數，$V_e(t)$表示抖音範圍參數，而抖音頻率參數 $V_r(t)$可以經由對$\phi(t)$作微分來得到，也就是

$$V_r(t) = \frac{1}{2\pi} \cdot \frac{\mathrm{d}\phi(t)}{\mathrm{d}t} \tag{2}$$

2.3. 音位軌跡之分析

估計音位軌跡 $V_d(t)$的一個基本的方法是，對 $f(t)$作低通濾波處理，實作上低通濾波可以在頻域或時域上進行，當我們在頻域上作低通濾波的處理時，發現了一個嚴重的問題，就是 IPF 曲線和音位軌跡之間的差值，$f(t)$ - $V_d(t)$，會在曲線左右兩端的時間點附近，變得很大，因此後來我們改成使用時域上的 moving average 的低通濾波作法，如此就免除了前述的問題。在時間點 t 時，我們以計算 $f(\tau)$, τ = t-128, t-127, …, t+128, 的平均值，來作爲 $V_d(t)$的值。

2.4. 抖音範圍和頻率之分析

令 $s(t)$ 表示公式(1) 裡的$V_e(t) \cdot cos(\phi(t))$，則我們可以計算 $f(t)$ - $V_d(t)$來求得 $s(t)$ 。對於 $s(t)$，接著使用解析信號之分析方法[13]，就可以將 $V_e(t)$和$\phi(t)$求取出來。

依據 Gabor 的定義[13]，令 $s(t)$的解析信號爲 $z(t)$，則 $z(t)$ 的建造方式是，以 $s(t)$作爲實部，而以 $\hat{s}(t)$ 作爲虛部，也就是

$$z(t) = s(t) + \mathrm{j} \cdot \hat{s}(t) \ , \tag{3}$$

$$\hat{s}(t) = H\left[s(t)\right] = \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{s(\tau)}{t-\tau} d\tau \ , \tag{4}$$

上式中， $H[s(t)]$表示 Hilbert 轉換，Hilbert 轉換能夠把信號的相位角恰好旋轉π/2 或-π/2，所以可以得到 $\hat{s}(t) = V_e(t) \cdot sin(\phi(t))$ ，再依公式(3)裡 $z(t)$的定義，推得 $z(t) = V_e(t) \cdot exp(\mathrm{j} \cdot \phi(t))$ ，此時就可依據這個 $z(t)$公式，來求得 $V_e(t)$和$\phi(t)$，也就是，

$$V_e(t) = \sqrt{s^2(t) + \hat{s}^2(t)} \quad , \tag{5}$$

$$\phi(t) = arc\,tan\left(s(t), \hat{s}(t)\right) , \tag{6}$$

得到$\phi(t)$後，再依公式(2) 作計算，就可以得到抖音頻率 $V_r(t)$ 的曲線。

關於公式(4)的Hilbert轉換之計算，實作上我們參考了 Suzuki[14]和 Langton[15]等人的論文，而採取了如下的作法: (a)對N個樣本的整段信號作一次離散傅立葉(DFT)運算，以得到N個頻率點的long-term頻譜；(b)調整頻譜，將編號在0至N/2之間的頻率點上的頻譜振幅值乘以2倍，但是把編號在N/2至N-1之間的頻率點上的頻譜振幅值強迫設爲0值；(c)作反向DFT運算，還原成時域信號，令所得的信號序列爲C[n]=A[n]+jB[n]，則A[n]會是原始的信號序列，而B[n]會是Hilbert 轉換後的信號序列，即C[n]相當於公式(3)的$z(t)$。

3. ANN 抖音參數模型

在本研究裡，我們使用 ANN 模型來學習真人歌者在抖音表達方面的歌唱風格。由於前一節裡，對於一個歌聲音節，可以分析出音位軌跡 $V_d(t)$、抖音範圍 $V_e(t)$、抖音頻率 $V_r(t)$、和初始相位$\phi(0)$等 4 種抖音參數，因此我們決定對各類型的參數分別去訓練、建立各自的 ANN 模型，實際上這些 ANN 都是 MLP (multi-layer perceptron) [16]，而在此所用的學習演算法是倒傳遞(back propagation)法。

各個 MLP 的結構如圖 3 所示，也就是在輸入層和輸出層之間，放置了一層的隱藏層，此外在隱藏層和

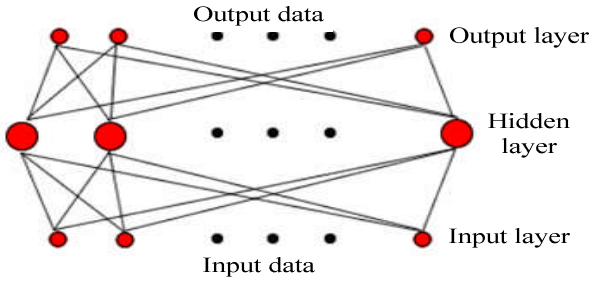

圖 3 MLP 之結構

輸出層的各個節點裡，我們使用的轉換函數是雙曲線正切函數，其定義爲

$$f(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}} \quad , \tag{7}$$

這是因爲抖音參數的數值可能爲正值或負值。關於 MLP 的輸出層，在節點數方面，其中三種抖音參數的 MLP，我們設定爲 32 個，而對於初始相位的 MLP，則只需使用一個；至於抖音參數的表示和正規化的作法，則會在 3.1 節裡討論。關於 MLP 的輸入層，它是用來接收目前要唱的音節的語境資訊，我們所定義的語境資訊的項目，將會在 3.2 節裡說明。

3.1. 抖音參數取樣和正規化

各個歌聲音節被演唱的時間長度，會因爲歌曲速度(tempo)和拍數(beats) 的不一樣，而發生相互之間有很大的差異存在，也就造成了不同音節所分析出的抖音參數曲線(如音位軌跡的曲線)，時間長度上會是參差不齊的。因此，我們必需使用一種適當的表示方式，把不等長度的曲線轉成固定維度(dimension)的目標數值，以讓 MLP 去學習。在本論文裡我們採用了一種簡單的表示方式，就是在一條曲線的時間軸上，均勻放置 32 個取樣點，去對曲線作取樣(sampling)。如此，一條抖音參數的曲線 $V_x(t)$，就會被取樣出 32 個樣本值，$U_x(i) = V_x(T*i/31)$, i=0, 1, …, 31, 其中 T表示曲線的時間長度。

從相反方向來看，當我們從一個 MLP 的輸出層取得 32 點的抖音參數表示值 $U_x(i)$，並且被規定一個時間長度值 T，則要如何去合成出這個抖音參數的曲線? 一個基本的想法是透過內差的處理，在本研究裡，我們採取了片段線性(piece-wise linear)內差的作法，其效果似乎還不錯。詳細的作法是，對於一個時間點 t，我們首先尋找出包含 t的時間區間[T_i, T_{i+1}), $T_i = T \bullet i / 31$，然後在 t 時間的曲線值 $V_x(t)$，就可以如下的線性內差公式來求得。

$$V_x(t) = U_x(k) + \left(U_x(k+1) - U_x(k)\right)\frac{t - T_k}{T_{k+1} - T_k} \tag{8}$$

在訓練 MLP 時，上述取樣得到的 32 點的曲線樣本值 $U_x(i)$，並不可以直接作爲 MLP 學習的目標值，這是因爲公式(7)所定義的轉換函數的值域只有在-1 到 +1 之間，爲了配合這樣的數值範圍，曲線的取樣值 $U_x(i)$，必需先作正規化的處理。令 $U_d(i)$是音位軌跡 $V_d(t)$的取樣值，則我們要先去計算一個正規化因數 M_d，它是由 $U_d(i)$的中段部分的數值去作幾何平均而求得，詳細的公式是

$$M_d = \left(\prod_{i=11}^{20} U_d(i) \right)^{1/10} \quad , \tag{9}$$

前段和後段的數值未被使用到，那是因爲該部分的數值可能因爲轉音(portamento)而發生不穩定的情況。計算出 M_d後，正規化的取樣值就可以如下公式來計算。

$$\hat{U}_d(i) = \frac{U_d(i)}{M_d} - 1, \quad i = 0,1,...,31 , \tag{10}$$

令 $U_e(i)$是抖音範圍曲線 $V_e(t)$的取樣值，在此採取的正規化作法是，將 $U_e(i)$ 除以 $U_d(i)$，也就是 $\hat{U}_e(i) = U_e(i) / U_d(i)$。此外，抖音頻率曲線的取樣值 $U_r(i)$的正規化作法是，將它們直接除以常數 20，也就是 $\hat{U}_r(i) = U_r(i) / 20$。至於初始相位$\phi(0)$，我們直接將它除以常數 5 來作正規化。

3.2. 語境資訊及其分類

那些因素會影響抖音的表達? 我們覺得可能的因素包括: (a)本次要唱的音節的時長(duration)、聲母類別、韻母類別；(b)前一個音節的時長和韻母類別；(c)下一個音節的時長和聲母類別；(d)本音節和前後音節之間的音程差距。由於我們要納入考慮的因素不少，這些因素的數值的可能組合的數量，將會是非常龐大，相對地我們目前收集到的訓練用的歌曲僅有 15 首，以音節來算則只有 2,841 個音節。因此我們不得不對這些因素的數值作分類，以便讓這些因素的可能組合的數量能夠大幅下降。

在 3 個時長的因素之中，我們認爲本音節的時長，要比相鄰兩音節的時長來得重要，因此我們把本音節的時長分成 5 類，而只把前後音節的時長分成 3 類。本音節時長的 5 個分類的設定是: 0~0.3 秒，0.3~0.5 秒，0.5~0.8 秒，0.8~1.3 秒，和 1.3 秒以上。至於前後音節時長的 3 個分類的設定是: 0~0.25 秒，0.25~0.5 秒，和 0.5 秒以上。依據前述的分類數量，所以它們各別需要 3bits 和 2bits 來表示不同的時長分類。

在二種韻母因素的分類上，我們把國語的 39 種韻母分成 4 個分類，也就是分成單母音韻母(如/a/)、雙母音韻母(如/ai/)、三母音韻母(如/iau/)、和鼻音尾韻母(如/ang/)。另外，在二種聲母因素的分類上，我們把國語的 21 種聲母分成 3 個分類，也就是分成有聲聲母(如/m,r/)、短無聲聲母(如/b,z/)、和長無聲聲母(如/p,s/)。如此，韻母和聲母的分類，就各別都需要 2bits 去表示。

在本音節和前後音節的音程差方面，相鄰音符的音程差距這裡以半音(semitone)爲單位來計算，其差異值的範圍將會比-12 ~ +12 還大，因此我們把音程差值分成 7 個分類，詳細的分類方式如表 1 所示。要表示 7 個分類，所以需使用 3bits。

表 1 音程差值的分類方式

Class	1	2	3	4	5	6	7
Elements	-6,-7, -8,…	-3,-4, -5	-1,-2	0	1, 2	3, 4, 5	6, 7, 8,…

3.3. 模型訓練之實驗

關於隱藏層裡的單元數的設定，我們作了一些 MLP 訓練的實驗。我們令初始學習速率爲 2.0，學習折減因子爲 0.95，且設定訓練循環回數爲 1500，在如此條件下分別對隱藏層單元個數 6、8、10、12、16、24 等作測試，以觀察不同單元數的影響，結果發現設爲 8 個單元時，總體來說訓練誤差是比較小的。以音位軌跡之 MLP 模型的訓練爲例，對於不同的隱藏層單元個數，我們量測得到的誤差，就如表 1 裡所示的數值。

表 2 音位軌跡 MLP 模型訓練之誤差量測

單元數	Avg rms err.	Std rms err	Max rms err
6	0.037356	0.033735	0.331159
8	0.037544	0.033663	0.331511
10	0.039361	0.033945	0.324763
12	0.03875	0.033752	0.328735
16	0.039405	0.034365	0.328274
24	0.039321	0.034112	0.333483

4. 歌聲信號合成與聽覺測試

第 2、3 節敘述了 MLP 抖音參數模型的建造方法，接著我們利用此模型來製作一個國語歌聲的合成系統，這個系統的主要處理流程如圖 4 所示。每次從歌譜檔案輸入一個音符；然後依據速度和拍數資訊來計算出此音符的時長；接著把語境資訊整理、編碼成 MLP 需求的輸入格式，以帶入 MLP 去產生出具有抖音表達的基週軌跡，產生方式將在 4.1 節詳細說明；之後在圖 4 的最後方塊裡，依據基週軌跡去調整歌詞音節的 HNM 參數數值，再使用一個改進的 HNM 爲基礎的合成法，去合成出歌聲信號，較詳細的說明在 4.2 節裡。

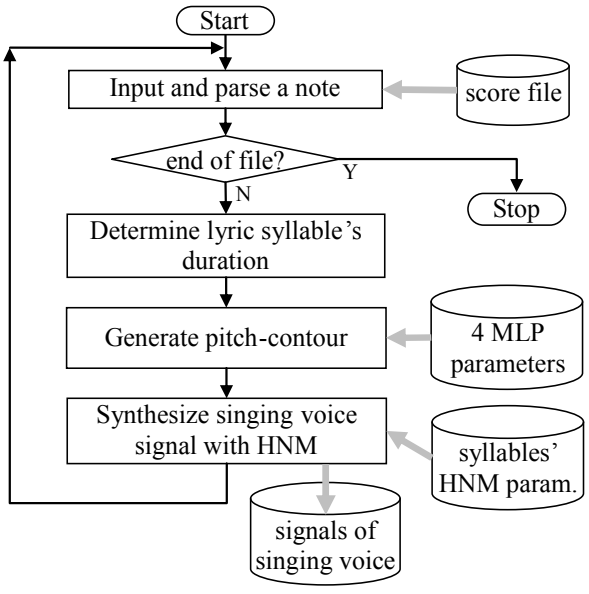

圖 4 國語歌聲合成之主要處理流程

4.1. 基週軌跡產生

當帶入語境資料到 MLP 後，MLP 輸出的是正規化的抖音參數的取樣值，因此要作反正規化的計算，以得到正確尺度的取樣值，$U_d(i)$，$U_e(i)$，$U_r(i)$，和$\phi(0)$。關於音位軌跡曲線的產生，我們需先知道本音節所唱音符的音高頻率值 F (Hz)，它可由音符的音高符號(如"G3")去推算。此外，我們還需要知道本音節的時長值 T，這已在圖 4 裡的第二個方塊計算出來。接著把公式(10) 裡的 M_d 以 F 取代，再把 MLP 輸出的 $\hat{U}_d(i)$ 拿去作反向計算，即$U_d(i)$=$(\hat{U}_d(i)+1)\cdot F$ ，就可得到具有正確音高的 $U_d(i)$，然後依據時長值 T 和公式(8)作內差，就可得到音位軌跡曲線 $V_d(t)$。

關於抖音範圍的取樣值 $U_e(i)$，其求取方式是把 MLP 輸出的正規化值 $\hat{U}_e(i)$ 乘上 $U_d(i)$，即 $U_e(i) = \hat{U}_e(i)\cdot U_d(i)$，接著依據時長值 T 和公式(8)作內差，即可得到抖音範圍曲線 $V_e(t)$。類似地，可把 $\hat{U}_r(i)$ 乘上 20 來得到 $U_r(i)$，然後依據時長值 T 和公式(8)作內差，來求得抖音頻率曲線 $V_r(t)$。

在求得 3 種抖音參數的曲線後，接著先使用 $V_r(t)$ 曲線來計算相位曲線$\phi(t)$，計算公式爲

$$\phi(t)=\phi(t-1)+2\pi\cdot V_r(t)\cdot\frac{1}{22{,}050},\quad t=1,2,...,T-1 \qquad (11)$$

上式中 22,050 是取樣率。算出相位曲線後，基週軌跡的曲線 $P(t)$就可以如下公式來計算得到。

$$P(t)=V_d(t)+V_e(t)\cdot cos\left(\phi(t)\right),\quad t=0,1,...,T-1 \qquad (12)$$

當我們以公式(11)和(12) 去計算歌曲"快樂頌"裡的第一個樂句 /mi, mi, fa, sol, …, mi, re, re /時，所得到的基週軌跡如圖 5 所示，由此圖可發現，各個音符合成的基週軌跡都是有起伏的，且最後的/re/更有明顯的週期性擺動，亦即抖音現象。我們認爲這樣的現象(各個音符都有起伏)是正常的，因爲人類唱歌時不會故意要唱出平平的基週軌跡，至於起伏是大或是小? 人眼的觀察會受縱軸(頻率軸)尺度的影響，而應以人耳的感受來判斷，實際上圖五的基週軌跡聽起來是頗爲自然的。

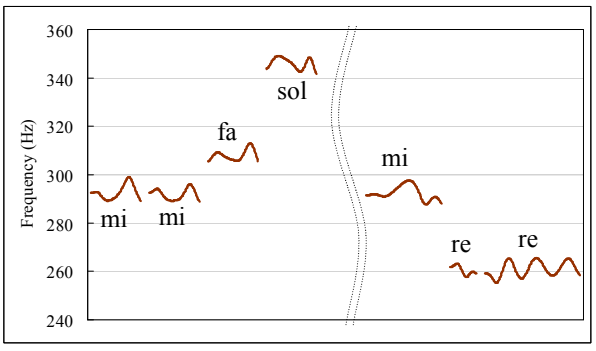

圖 5 基週軌跡合成之例子

4.2. 歌聲信號合成

國語的一個特性是，只有 408 種不同的音節，因此我們邀請一位女性到 RE-242 隔音室，來錄國語的 408 種音節的發音，不過各種音節都只有錄、存一次發音，並且這裡的錄音者和錄 MLP 訓練歌曲的是不同人。由於各種音節只有一次發音，所以我們不能夠作單元選擇；此外一個音節所分析出的 HNM 參數，必需被用來合成出種種音高、時長組合的歌聲信號。

要在前述的限制之下達成目標，如何維持音色(timbre)的一致性，是一個必需解決的問題，不然，欲合成的基週軌跡和原始錄音裡的基週軌跡差異很多時，音色可能就被改變了；此外，當欲合成的時長和原始錄音裡的時長差異很多時，如何合成出流利(fluent)發音的歌唱聲，也是一個需要考慮的問題。前述的兩個問題，我們已經在先前的研究裡提出可行的解決方法[6]，並且原始提出 HNM 的作者 Stylianou 的論文[11, 12]，也是可以參考的，因此我們就不再詳細敘述 HNM 爲基礎的人聲信號合成方法。

簡短來說，歌聲信號看成是由兩種信號成分相加得到，一個是佔據低頻帶的諧波信號 $H(t)$，另一個是佔據高頻帶的雜音信號 $N(t)$。$H(t)$的合成公式是

$$H(t)=\sum_{k=0}^{L}a_k^n(t)cos\left(\phi_k^n(t)\right),\quad t=0,1,...,T^n, \qquad (13)$$

其中 L 表示諧波成分的數目，T^n 是介於第 n 和第 $n+1$ 個控制點之間的信號樣本數，$a_k^n(t)$ 是第 k 個諧波在時刻 t 的時變振幅，$\phi_k^n(t)$ 則是第 k 個諧波在時刻 t 時的累積相位。此外，雜音信號 $N(t)$的合成，在高頻帶上先分別產生間隔 100Hz 的弦波信號，再去作加總，但是這裡的各個弦波的頻率是固定的。

4.3. 聽覺測試

我們使用同一個歌譜檔案，來分別合成出三個歌聲檔案，第一個歌聲檔以 *SA* 表示，是以沒有抖音的方式作合成，詳細作法是令 $U_d(i)$=F 和 $U_e(i)$=0；第二個歌聲檔以 *SB* 表示，是使用固定的抖音參數數值來合成，也就是設定 $U_d(i)$=F、$U_e(i)$=F*3/100、和 $U_r(i)$=4；第三個歌聲檔以 *SC* 表示，先使用 MLP 來產生出四種抖音參數的數值，再據以作抖音歌聲的合成。

接著，我們把這三個歌聲檔依 *SA*、*SB*、*SC* 之次序分別播放給 15 位參與聽覺測試者聆聽，然後請各個聽測者給二個自然度比較的評分，一個是比較 *SA* 和 *SB*，另一個則是比較 *SA* 和 *SC*。評分的方式是，當兩歌聲檔之間的自然度無法區分時，給 0 分；當後者(前者)比前者(後者)稍好一些時，給 1 (-1)分；另外，當後者(前者)比前者(後者)稍好很多時，給 2 (-2)分。

依據 15 位聽測者所給的評分，我們計算出的平均分數分別是，0.63 分 於比較 *SA* 和 *SB* 時，和 1.29 分於比較 *SA* 和 *SC* 時。這樣的評分結果顯示，使用 MLP

產生的抖音參數所合成出的歌聲，其自然度的確可以獲得明顯的改進。爲了展示本研究所合成出的歌聲，我們設置了一個網頁，來供有興趣者去瀏覽及下載歌聲檔試聽，其網址爲 http://guhy.csie.ntust.edu.tw/vibrato/。

5. 結論

真人歌唱時，抖音是最常被使用來傳達歌聲表情的一項技巧，因此對於電腦歌聲合成的研究來說，抖音風格的塑模(modeling)和抖音參數的生成是必需考慮的議題。在本論文裡，我們研究了抖音參數(音位軌跡、抖音範圍、抖音頻率)曲線的分析、和表示的方法，然後提出、實驗以 MLP 來建立抖音參數的模型。

由實驗的結果來看，從短時傅利葉轉換的頻譜去偵測瞬間音高頻率，再以解析信號的方法來分析出抖音參數的曲線，這樣的程序的確是可行且有效的作法。此外，我們以 32 點取樣值來作片段線性內差，以逼近一條抖音參數曲線；並且研究了抖音參數正規化的作法，以輸入 MLP 作訓練；然後以 MLP 輸出的抖音參數，去控制 HNM 作歌聲信號合成。依據聽測實驗的結果，我們可說，前述的處理步驟及以 MLP 建立的抖音參數模型，的確可用以合成出較爲自然的歌聲信號。未來我們將錄製更多的訓練用之歌曲，以進一步研究、觀察 MLP 抖音參數模型的效能。

6. 誌謝

感謝國科會對本研究的支援，計畫編號爲: NSC-96-2218- E-011-002.

7. 參考文獻

[1] F. R. Moore, *Elements of Computer Music*, Prentice-Hall, 1990.

[2] C. Dodge and T. A. Jerse, *Computer Music: Synthesis, Composition, and Performance*, second edition, Schirmer Books, New York, 1997.

[3] M. W. Macon, L. Jensen-Link, J. Oliverio, M.A. Clements and E.B. George, "A Singing Voice Synthesis System Based on Sinusoidal Modeling", Proceedings of *IEEE ICASSP*, Vol. 1, pp. 435-438, 1997.

[4] N. Schnell, G. Peeters, S. Lemouton, P. Manoury, and X. Rodet, "Synthesizing a Choir in Real-Time Using Pitch Synchronous Overlap Add", Proceedings of *International Computer Music Conference*, pp. 102-108, 2000.

[5] J. Bonada and A. Loscos, "Sample-based Singing Voice Synthesizer by Spectral Concatenation", Proceedings of *The Stockholm Music Acoustics Conference*, Stockholm, Sweden, Aug. 2003.

[6] H. Y. Gu and H. L. Liau, "Mandarin Singing Voice Synthesis Using an HNM Based Scheme", to appear in Proceedings of *International Congress on Image and Signal Processing*, Sanya, China, May 2008.

[7] Y. Horii, **"**Acoustic Analysis of Vocal Vibrato: a Theoretical Interpretation of Data**"**, *Journal of Voice*, Vol. 3, pp. 36–43. 1989.

[8] S. Imaizumi, H. Saida, Y. Shimura, and H. Hirose, **"**Harmonic Analysis of the Singing Voice: Acoustic Characteristics of Vibrato**"**, Proceedings of *The Stockholm Music Acoustics Conference*, Royal Swedish Academy of Music, Stockholm, pp. 197–200. 1994.

[9] J. Sundberg, E. Prame, and J. Iwarsson, **"**Replicability and Accuracy of Pitch Patterns in Professional Singers**"**, *Vocal Fold Physiology*, edited by P. J. Davis and N. H. Fletcher, Singular, San Diego, 1996.

[10] J. I. Shonle and K. E. Horan, "The Pitch of Vibrato Tones", *J. Acoust. Soc. Am.*, Vol. 67, pp. 246–252. 1980.

[11] Y. Stylianou, *Harmonic plus Noise Models for Speech, Combined with Statistical Methods, for Speech and Speaker Modification*, Ph.D. thesis, Ecole Nationale Supèrieure des Télécommunications, Paris, France, 1996.

[12] Y. Stylianou, "Modeling Speech Based on Harmonic plus Noise Models", *Nonlinear Speech Modeling and Applications*, Springer-Verlag, Germany, 2005.

[13] H. G. Feichtinger and T. Strohmer, *Gabor Analysis and Algorithms: Theory and Applications*, Birkhauser, Boston, 1998.

[14] H. Suzuki, et al., "Instantaneous Frequencies of Signals Obtained by the Analytic Signal Method", *Acoust. Sci. & Tech.*, Vol. 27, pp. 163-170, 2006.

[15] C. Langton, "Hilbert Transform, Analytic Signal and the Complex Envelope", *Local Space Systems*, http://www.complextoreal.com/tcomplex.htm..

[16] K. Gurney, *An Introduction to Neural Networks*, UCL Press, 1997.

[17] 詹詩涵，基於音高調節之歌聲合成系統，碩士論文，資訊系統與應用研究所，國立清華大學，2006。

[18] Wen-Hsing Lai, "A Mandarin Singing Synthesis System", 2007 Int. Workshop on Computer Music and Audio Technology (WOCMAT2007), Hsin-Chu, Taiwan, Session II (Sound Synthesis), 2007.