使用頻譜HMM模型及波形包絡模型之曲笛聲合成

Chinese-flute Sound Synthesis Using Spectral-HMM and Waveform-envelope Models

古鴻炎
台灣科大資訊工程系
guhy@mail.ntust.edu.tw

曾聖文
台灣科大資訊工程所
m9915066 @mail.ntust.edu.tw

摘要

本論文使用 HTS 的頻譜 HMM 模型及自行建立的波形包絡模型，來製作一個曲笛聲音的合成系統。首先我們以 STRAIGHT 軟體來分析各個曲笛樂句錄音的基頻軌跡，再自行發展程式作音符之自動標音；接著，使用 HTS 軟體來訓練頻譜 HMM 模型及決策樹；然後，對各音符的波形包絡作 DCT 轉換，再以前後文分類後算出的平均 DCT 向量作為包絡模型。在合成階段，先令 HTS 軟體作笛聲合成，然後以我們程式產生的基頻軌跡去取代 HTS 所產生的，再以包絡模型 DCT 向量還原出的包絡曲線，去調整 HTS 再次合成的波形包絡，如此就可合成出音高正確且比較自然的曲笛聲音。之後我們進行頻譜誤差的量測、及主觀聽測的評估，實驗結果顯示，作基頻取代及波形包絡調整後的合成曲笛樂曲，其自然度會比 HTS 原始合成的好很多。

關鍵詞：樂器聲合成、HMM 頻譜模型、波形包絡。

1. 前言

疲勞的時候聽音樂，可以減少疲勞，緊張的時候聽音樂，可以紓解壓力，所以音樂可以陶冶心靈。隨著電腦科技日益進步，將音樂演奏的聲音加入些電腦產生的聲音，或是由電腦完全取代人力演奏，去產生出樂器聲音，未來將會是可實現的。

笛子是中國古老的吹奏樂器，是一種不斷被發展的主要獨奏與合奏樂器，依其長度及音域分成小笛、梆笛、曲笛、大笛。曲笛的音色粗曠、高昂、清脆、嘹亮，音調渾厚圓潤，柔美流暢。西方的音樂講究和聲，希望可以作到合奏並且和諧的境界，然而曲笛聲音與西洋音樂相反，它有如孤高的隱士，在獨奏時才可以顯現它絕世獨立的美。

我們從文獻回顧發現，樂器聲音合成的研究大部分使用的是西方樂器，研究東方傳統樂器聲音合成的則很少。並且，MIDI的標準音色中，沒有傳統笛子的音色，因此可推知，MIDI軟體很難產生出傳統笛子的音樂(如”陽明春曉”之樂曲)。我們覺得東方樂器聲音的優美程度不輸於西方樂器，所以選擇了曲笛來作研究，希望可以藉由電腦來合成出它的聲音。

HTS (HMMs trained by the HMM-based speech synthesis system) [12]為一套知名的語音合成軟體，由於其程式原始碼是開放的，所以被廣泛的用於語音合成的研究領域。但是，最近已經有人把它應用於小提琴聲音的合成上[3]，所以我們也想嘗試以HTS作基礎，來研究曲笛聲音的合成。

本論文的研究方法是，應用HTS良好的頻譜HMM (hidden Markov model)模型的建造功能，來為曲笛吹奏的各種音高的音符，建造一個對應的頻譜HMM模型[9]；並且透過HTS的決策樹功能，去挑選出最符合前後音符音高相關性的頻譜HMM。至於HTS不擅長(經常發生錯誤)的基頻軌跡與振幅包絡的產生，我們將自行發展產生相關參數的方法與程式模組，再把我們的模組和HTS作結合，以便合成出自然、且音色近似曲笛的樂器聲音。我們的曲笛聲音合成系統，可分為模型訓練與笛聲合成兩個階段。

2. 模型訓練

我們將訓練階段分為八個處理方塊，如圖1所示，這些方塊的功能為: (1)首先將錄好的曲笛音檔

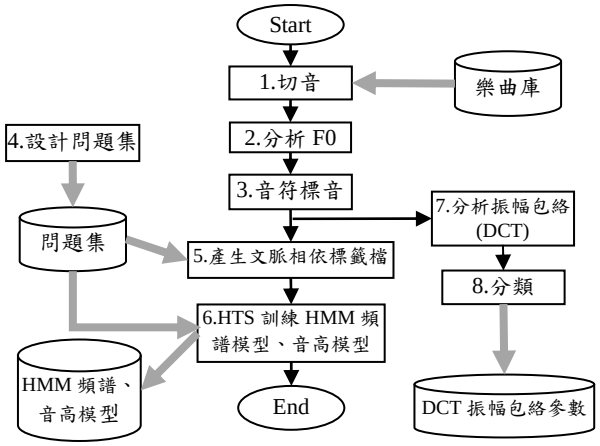

圖1 訓練階段之主流程

切成以樂句為單位之音檔，以利於使用STRAIGHT分析程式[7]；(2)接著，藉由STRAIGHT分析出各音檔的基頻軌跡(F0)；(3)依據樂譜資料及STRAIGHT分析出的基頻軌跡，以自行發展的程式進行音符的自動標音處理；(4)接著設計HTS決策樹的問題集；(5)根據所設計的問題集與音符的標音檔，寫程式去產生文脈相依標籤檔；(6)然後使用HTS去訓練頻譜HMM模型與音高HMM模型；(7)依據標音檔的資訊，去對各個音符的信號波形作離散餘弦轉換(discrete cosine transform, DCT)，以求得各音符的振幅包絡參數；(8)根據各音符的音高、音長參數去對

各音符的DCT振幅參數作分類，再對各類別的DCT參數去計算出平均DCT向量與標準差。

2.1 錄音、標音

我們邀請本校國樂社一位有多年曲笛吹奏經驗的同學來錄音，錄音的地點在本系的隔音錄音室(Acoustic Systems RE-242)所錄製，電腦外接的錄音介面為M-Audio公司出品的Fast-Track設備，並使用RODE麥克風NT2-A來收音，音檔的取樣率為44,100 Hz、解析度為16 bits/sample。選取樂曲時，把快慢節奏的樂曲都加入，希望在合成階段時，可以掌握不同快慢節奏的樂曲。我們總共錄製了20首樂曲，切割後分成530個樂句，共有6170個音符。

為了擷取一個樂句裡各個音符的信號，必須先作標音的動作，就是在時間軸上標示各個音符的左右邊界與音符音名。實作上，我們先以STRAIGHT分析出基頻資料，再自行撰寫程式，依基頻資料去偵測出一個樂句裡的各個音符，然後輸出成標籤檔。接著，再使用Wavesurfer軟體來對音符邊界作人工之微調、更正，此為文脈無關之標籤檔。

2.2 HMM 模型訓練

HTS [14]是由日本名古屋工業大學(Nagoya Institute of Technology)的Keiichi Tokuda等人所開發的基於HMM的語音合成軟體，其核心組件是取自HTK (Hidden Markov Model Toolkit) [13]，HTS的主要目的是讓人可以快速的開發一套語音合成系統。

HTS軟體訓練HMM模型的流程，我們仿效[2]的畫法，依據HTS的命令檔畫出圖2之流程。本論文使用HTS網站提供的Demo批次命令檔來進行訓練，不過因為語料不同且原Demo命令檔是作歌聲合成的，所以我們必須修改一些相關的參數。

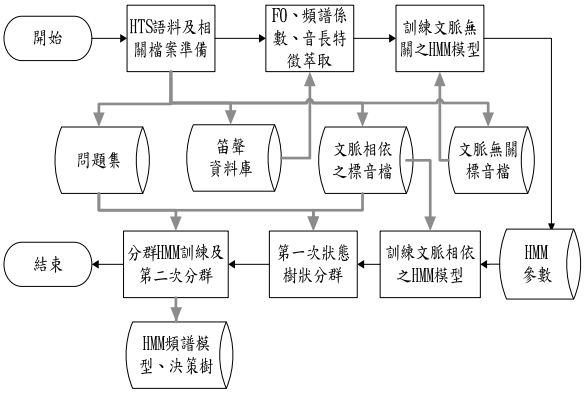

圖2 HTS訓練HMM的流程

訓練頻譜HMM模型的步驟如下：第一步，先將自備的曲笛聲音檔及標籤檔準備好；第二步，對各個笛聲樂句音檔作頻譜係數、基頻軌跡及音長特徵的擷取；第三步，讀入文脈無關之標音檔，訓練文脈無關之HMM模型；第四步，將文脈無關之HMM模型依據文脈相依之標音檔作拓展，訓練出文脈相依的HMM模型；第五步，對於文脈相依HMM模型，作樹狀分群，分群時以狀態為單位，分別產生HMM各狀態之決策樹；第六步，對於第五步分群的結果，進行HMM訓練，之後進行第二次的樹狀分群，分群時以狀態為單位，分別產生HMM各狀態之決策樹。

2.3 HTS參數設定與輸入檔案準備

本論文使用2.2版之HTS軟體，並且採用Demo套件"HTS-demo_NIT-SONG070-F001"中的訓練階段命令檔去訓練HMM模型及決策樹。

2.3.1 笛聲單元與HMM模型

我們準備曲笛音檔的方式，已在2.1節說明，所錄製的曲笛樂曲，音符的音高共有18種，如表1所列，因此當以音高作區分時，就需要建造18個HMM模型。

表 1 曲笛音符之音名與對應的頻率

音名	頻率值(HZ)	音名	頻率值(HZ)
A4	440.00	C6#	1,108.7
B4	493.88	D6	1,174.7
C5#	554.37	E6	1,318.5
D5	587.33	F6#	1,480.0
E5	659.26	G6	1,568.0
F5#	739.99	A6	1,760.0
G5	783.99	B6	1,975.5
A5	880.00	C7#	2,217.5
B5	987.77	D7	2,349.3

此外，我們統計了各類時長(duration)的音符個數，依時長的分佈情形，我們再將每一種音高的音符依時長值概分為2類，稱為長音與短音，當音符時長大於0.2秒就歸為長音，而小於0.2秒就歸為短音。如此當考慮長、短音的區分時，需要建造的HMM模型個數，就增為36個。

2.3.2 HTS參數設定與輸入檔案

HTS是在linux的環境下運作，在執行Demo的命令檔之前，需先修改參數設定檔，本論文的主要參數設定如表2所示。

表 2 HTS 的參數設定

參數名稱	值	說明
SAMPFREQ	44,100	設定取樣率為 44,100 Hz
GAMMA	0	0 代表逼近於 MFCC 係數。
FREQWARP	0.1	梅爾頻率軸曲度。
LNGAIN	0	使用 log 尺度的基週軌跡。
FRAMELEN	1,102.5	設定音框長度為 25ms。
FRAMESHIFT	220.5	設定音框位移為 5ms。
MGCORDER	34	擷取 34 維 MGC 係數。
LOWERF0	100	最低 F0 設定為 100。
UPPERF0	4,000	最高 F0 設定為 4,000。

在起動HMM的訓練程序之前，必須先準備好三種資料檔案，分別為:標籤檔(包含文脈相依與文脈無關兩種)、轉換為raw檔(不含wav音檔之檔頭部

份)形式之笛聲資料庫、及問題集。

2.3.3 頻譜係數萃取

本論文使用的頻譜特徵係數為MGC係數，MGC係數即為廣義梅爾倒頻譜係數(Mel-generalized cepstrum, MGC)，其特色為可以透過調整參數數值而改變其使用的頻率軸刻度(如梅爾頻率刻度)，並且可逼近多種現有係數的特性。我們透過Demo命令檔的參數設定，去呼叫SPTK (Speech Signal Processing Toolkit) [11]工具軟體來擷取出MGC係數。

2.4 問題集

HTS會依照我們提供的問題去比對標籤檔裡的資料，並且依據MDL(minimum description length)[5, 10]準則去建立二元決策樹，將文脈相近的HMM歸為一類。我們先依據音符的音高、音長去作分類，然後才用於設計問題集。

音高部份: 對於笛聲樂曲來說，除了本音符的音高資訊外，與前、後音符的相對音高也是重要的資訊。因此設計問題集時，除了把每一種音高獨立列為一個類別，也把相鄰音符的音高差值加入問題集中，以m19 ~ p19來表示，m代表負的差值，p代表正的差值，數字則是相差的半音(semitone)數。

音長部分:為了讓各音長類別平衡，我們依據音符時長的分佈情形，將音長分為四類。小於0.1秒的音長為第一類；0.1~0.2秒之間的音長為第二類；0.2~0.4秒之間的音長為第三類；大於0.4秒的音長為第四類。

此外考慮前後音符的相依性，我們再擴充兩個問題集，即為左邊音符相依問題集和右邊音符相依問題集，然後將這三個問題集合併後，就是本論文所使用的問題集。

問題集的一個範例如圖3所示，這是一個音符位置的相關問題，QS代表問題集，後面兩個雙引號中間為問題名稱，L代表當前音符左邊界交界之問題，大括號內的是問題的內容，每個問題以逗號隔開，星號/*/是萬用字元表示可以接任何的字元。圖3裡第一列的問題是，左邊接的音符是否為靜音(silence)，第二列的問題是，左邊接的音符音名是否為D5。

```
QS "L-Phone_silence"      {sil-*,sp-*}
QS "L-D5"                 {D5-*}
QS "L-E5"                  {E5-*}
QS "L-F5#"                 {F5#-*}
…
```

圖3 音符位置問題集範例

2.5 文脈相依的標籤檔格式

文脈相依的標籤檔，它的格式是根據預先設定好的問題集來作語法定義。我們自行撰寫程式，來讀取文脈無關的標籤檔，取得音長與音高音名的資料，再將各個資料按問題集的順序作編排，並在中間插入自行定義的連接符號，以完成整個文脈相依之標籤檔。圖4為我們所採用的文脈資料格式，符號"-"連接p1、p2，符號"+"連接p2、p3，這兩個符號為HTS規定的。符號"@"連接p3、a1，符號"^" 連接a1、a2，符號"~"連接a2、b1，符號"!"連接b1、b2，符號"#"連接b2、b3，這五個符號為自己定義的。

格式: p1-p2+p3 $A:@a1^a2 $B:~b1!b2#b3 定義: P1：前一個音符 p2：目前的音符 p3：後一個音符	a1：與前一個音符差幾個半音 a2：與後一個音符差幾個半音 b1：前一個音符的音長 b2：目前音符的音長 b3：後一個音符的音長

圖4 文脈相依標籤檔之格式與定義

一個文脈相依標籤檔的例子如圖5所示，當前的音高音名以粗體字標記，由此可知第一列的音高音名為sil，第二列為A4，第三列為A5。每一列的第一欄數字表示起始時間，第二欄數字表示結束時間，而第二欄數字減去第一欄數字代表音長，時間的單位為10^{-7}秒。以第二列為例來看，sil為前一個音符，A5為後一個音符，因為前一個音為靜音所以與它的音程差記為xx，m12為與後一個音的音程差，第一個4表示前一個音符的音長分類為4，第二個4表示後一個音符的音長分類為4，1表示目前音符的音長分類為1。

```
0 7560000 xx-sil+A4@xx^xx~xx!4#1
7560000 7640000 sil-A4+A5@xx^m12~4!1#4
7640000 15914185 A4-A5+A5@p12^p0~1!4#3
15914185 19540000 A5-A5+F5#@p0^p3~4!3#3
...
88620000 93400000 D5-sil+xx@xx^xx~4!4#xx
```

圖5 文脈相依的標籤檔例子

2.6 決策樹

HTS建造決策樹時，會將2.4節中的問題集當作非終端節點的候選集合，使用MDL準則選出若干問題作為非終端節點的決策內容，並且將HMM模型作為終端節點，來建立一棵二元決策樹。

一個HMM狀態的頻譜特徵決策樹的例子，如圖6所示，其中QS開頭的部份表示會被使用到的問題，L-E5表示問題名稱，後面大括弧中的值則是問題的內容，而"{*}{2}"下方就是決策樹，第一欄的0、-1、-8表示結點編號，後一欄為問題名稱，也是結點名稱，接著是下一個節點的位置，若出現如"mgc_s2_1"之字串則表示到達終端節點，已經找到所需要的模型；若第二欄問題的答案為是，則走右邊，否則走左邊。

3. 笛聲合成系統

我們研製的曲笛聲合成系統如圖7所示，主流

程分成六個處理方塊，各方塊的功能為：方塊1分析輸入的樂譜檔及產生出HTS需求的文脈相依標籤檔；方塊2令HTS合成出首次的笛聲信號，且產生

```
QS L-E5 { "E5-*" }
QS L-D5 { "D5-*" }
…
{*}{2}
{
   0      C-Phone_silence     -2      -1
   -1     C-sil               -23     -13
   …
   -8     C-A4                -9      "mgc_s2_1"
   …
}
```

圖6 HMM狀態的頻譜特徵決策樹例子

length.txt檔案；方塊3依據length.txt檔案存的音框數目記載，計算各個音符之時間位置的音框編號，然後以自製的程式去產生各音符的音高軌跡，再用以取代HTS產生的各音符音高軌跡；方塊4令HTS合成引擎依據文脈相依標籤檔取出各音符對應的HMM頻譜模型和取代過的音高軌跡去合成出信號波形；方塊5依樂譜資料取出各音符的DCT振幅參數，用以產生振幅顫動軌跡；方塊6對合成出的信號波形作振幅顫音調整，最後輸出音檔。

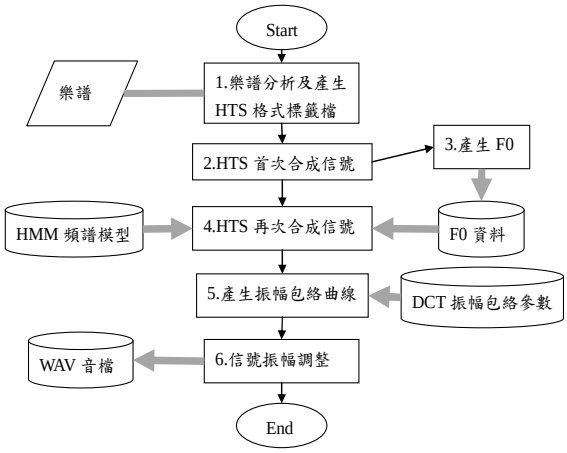

圖7 笛聲合成之主流程

3.1 樂譜檔處理

我們訂定了一種簡式樂譜檔的格式，以方便人工輸入一個樂曲的樂譜資料。簡式樂譜檔包含下列幾個部分：曲名、節奏拍數、滿度參數，接著是一序列音符的音名與拍數，一個例子如圖8所示，節奏拍數是每分鐘270拍，滿度值是0.95。

```
青春舞曲   270   0.95
F5#        1
E5         1
C5#        1
D5         1
…
```

圖 8 一個簡式樂譜檔的例子

節奏拍數可以控制整首樂曲的速度，因為每個音符都有一個對應的拍數，當配合節奏拍數，就可換算出這個音符要吹多少秒的時間。此外，王如江先前的研究[1]指出，音符的滿度是一項重要的參數，可用以控制合成的樂曲的綿密程度，滿度是指一個音符實際發聲的時間長度，佔該音符規定時長的比例。

3.2 文脈相依標籤檔與 HTS 信號合成

當輸入前述的簡式樂譜檔之後，我們以自製的字串處理程式，將簡式樂譜轉換成如圖5格式的文脈相依標籤檔，接著把標籤檔放入labels\gen資料夾內，再到scripts資料夾下找出Config.pm檔案，編輯Config.pm檔案，將訓練用的控制參數設為0以跳過HMM訓練的動作，而只保留合成部分的控制參數，因為HMM模型之訓練已在第2節完成。接著在訓練好的HMM模型資料夾內執行"make"，就可令HTS合成出曲笛聲音檔。

3.3 基頻軌跡重新產生

HTS所產生的基頻軌跡會有錯誤，例如某些有聲的音框會變成無聲，某些音框的基頻會發生半頻錯誤，此外曲笛的基頻抖動應該平穩，然而HTS產生的基頻軌跡卻會有劇烈的抖動。所以，我們對HTS合成引擎的程式碼(HTS_ pstream.c)作了修改，以讓HTS作首次信號合成時，把時長的相關資訊存入length.txt檔，我們再以自製的程式來產生新的基頻軌跡檔案reall.txt，然後在HTS_pstream.c檔案中加入讀取reall.txt的程式碼，就可以把原先HTS所產生的基頻值替換掉，這就是圖7區塊2與3分別要作的。

追蹤HTS_pstream.c檔可知，HTS只會把有聲音框對應的基頻值記錄起來，所以我們必需在HTS首次作信號合成時，把相關的資訊寫出到自行命名的length.txt檔案，該檔案每一列代表HMM一個狀態的資訊，第一個欄位表示此狀態為有聲或無聲；第二個欄位表示此狀態所駐留的音框數；第三個欄位則表示累積的音框總數。當一個HMM模型具有五個狀態時，則文脈相依標籤檔的每一列所標示的音符(或停頓)會相對應到length.txt檔的5列。如此，對照length.txt檔和文脈相依標籤檔，就可得知每個音符要產生出多少個音框的基頻值。

關於產生reall.txt檔的基頻值，因為曲笛音符的基頻軌跡大部分是穩定的值，所以每個音符的基頻值就可依其音高去查詢表1來取得對應的基頻值。不過，當相鄰兩音符之間的音程差距很大時，例如直接從A4跳B5，合成出的笛聲聽起來會有不順暢的感覺，因此在相鄰音符的交接處，我們加入了基頻值的弦波式內插[4]。準備好reall.txt檔之後，接著就可令HTS作第二次的信號合成，即圖7區塊4的處理。

3.4 合成之笛聲波形

在此令HTS使用9個狀態的HMM模型，並且合成出信號之前先對基頻軌跡作弦波式內插與基頻值取代。當HTS合成"茉莉花"樂曲之後，所合成出的笛聲信號如圖9所示，茉莉花屬於慢板，此為茉莉花樂曲的第三句，其旋律音符依序為A5、A5、A5、F5#、A5、B5、B5、A5。另外，當HTS合成"青春舞曲"樂曲之後，所合成出的笛聲信號如圖10所示，青春舞曲屬於快板，其旋律音符依序為F5#、E5、C5#、D5、F5#、E5、D5、C5#、B4、B4、G5、G5。

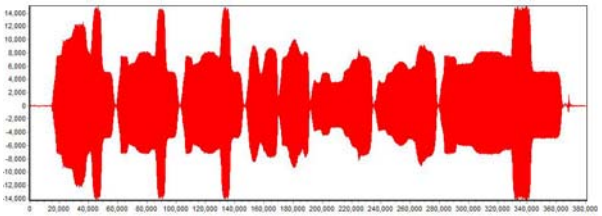

圖9 茉莉花之合成笛聲波形

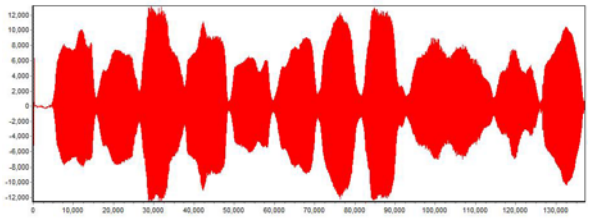

圖10 青春舞曲之合成笛聲波形

觀察圖10，可發現快板的青春舞曲，它各個音符的包絡曲線比較平順地變化，所以稱為自然的振幅包絡。但是觀察圖9，可發現慢板的茉莉花，它各個音符在波形外圍包絡上，劇烈變動而不自然。推測其原因是，HMM的訓練樂曲中，音符音長小於0.3秒的就佔了75%，而音符音長小於0.5秒的更含蓋了90%以上，然而圖9的茉莉花信號，每個音符的音長皆大於0.5秒，所以，導致包絡劇烈振動的原因是，長音音符的訓練資料不足。因此，在下一節我們研究了振幅包絡曲線的模型建造問題，以便對振幅包絡作調整修正。

4 振幅包絡調整

4.1 振幅包絡求取

關於振幅包絡的求取，我們研究的求取方法如圖11所示，包括直流支距(dc offset)扣除、整流、Gabor濾波[6]。

4.2 DCT 係數計算

求得一個音符的振幅包絡後，接著我們使用離散餘弦轉換(DCT)，將一個音符的振幅包絡曲線轉換成DCT係數。由於DCT轉換後可以把訊號的能量集中於低階的DCT係數，所以我們就只需抽取低階的少數個DCT係數來作為特徵係數。

文獻上DCT有多種公式，本論文採用最普遍的DCT-2 [8]，其公式如下:

$$C(m)=2\cdot\sum_{k=0}^{N-1}x(k)\cdot\cos[\frac{\pi\cdot m\cdot(2k+1)}{2N}],\ m=0,1,...,39, \quad (1)$$

上式裡$x(k)$表示振幅包絡，N為包絡的樣本點數，而$C(m)$就是DCT轉換後第m階的係數，依據實驗觀察，我們選擇使用40階的DCT係數。顛倒過來，從一組DCT係數去作反向DCT轉換，就可還原出原始包絡的逼近曲線，詳細的計算公式為:

$$x(k)=\frac{1}{N}\sum_{m=0}^{39}C(m)\cdot\cos[\frac{\pi\cdot m\cdot(2k+1)}{2N}],\ k=0,1,...,N-1, \quad (2)$$

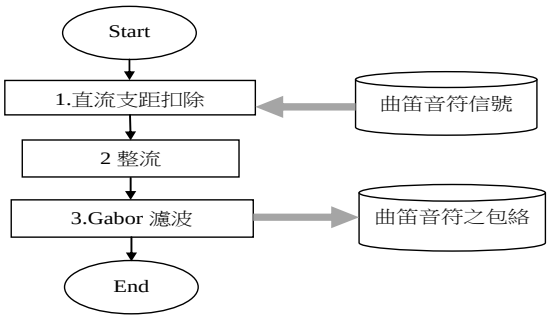

圖 11 振幅包絡之求取流程

4.3 振幅包絡分類

對於2.1節提到的笛聲錄音，我們先對所有音符求取各自的包絡曲線，再計算出各音符包絡的DCT係數。接著，對各個音符作分類，然後計算出各類別的平均DCT向量。之後在合成笛聲音符時，若一個音符的振幅包絡需作調整，就去計算此音符所屬的類別，以該類別的平均DCT向量來作反向DCT轉換，再依算出的逼近曲線去作振幅包絡的調整。所以本論文裡，我們初步地以一個類別的平均DCT向量，作為該類別的振幅包絡曲線的模型，將來會更進一步考慮振幅包絡模型建造的問題。

我們依據曲笛音符的音高和音長值來作分類。音高部分:在表1中列出曲笛音符的音高共有18種，在此我們假設鄰近的音高具有近似的振幅包絡，所以把每三種音高歸為一類，因此共分成六類音高。例如第一類為A4、B4、C5#，而第六類為B6、C7#、D7。音長部分:我們分別對相連的三個音符作分類，即前一個音符、目前音符及下一個音符個別的音長。音長分類的方式是，音長為0者(即沒有前接或後接音符)設為第一類，音長小於0.2秒者設為第二類，音長大於0.2秒且小於0.4秒者設為第三類，音長大於0.4秒者設為第四類。如此，共有6 × 4 × 4 × 4 = 384個振幅包絡類別。

4.4 信號振幅調整

我們自行發展程式來對笛聲音符的振幅包絡

作調整，此程式是對HTS合成出的笛聲信號作處理，而程式的流程如圖12所示，方塊1計算各個音符合成波形的振幅包絡，其方法已在4.1節裡以圖11之流程作說明；方塊2依據前、後接音符的音長及本音符音高去決定本音符的振幅包絡類別。

接著在方塊3裡，取出本音符所屬類別的40階DCT平均向量，再依本音符的音框總數N及DCT平均向量去還原出N個音框點的振幅包絡曲線，如果本音符的時長大於0.2秒，就以還原出的振幅包絡曲線去取代原本HTS合成信號的振幅包絡。

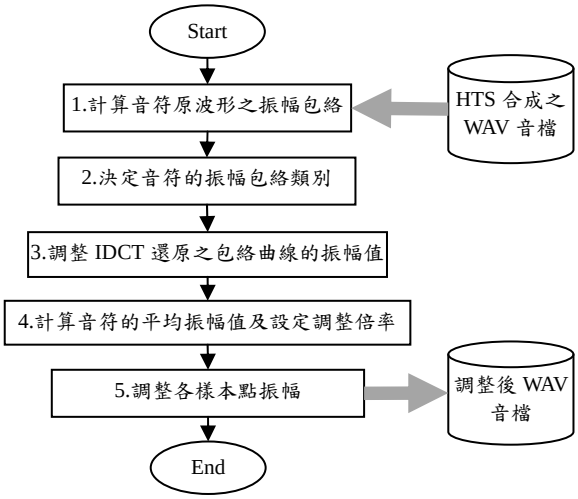

圖 12 曲笛聲信號振幅調整之主流程

圖12方塊4的動作是，計算本音符的平均振幅值，然後依據平均振幅來設定基準調整倍率，以使各音符之間的音量較有變化而變得活潑，因為我們觀察DCT平均向量所還原出的振幅包絡，音符之間音量幾乎沒有變化而顯得呆板。在圖12方塊5裡，對於相鄰音框之間的各個樣本點，先以線性內插方式求取調整倍率值，再將各個樣本點的振幅乘上它對應的基準調整倍率和短時調整倍率。

圖13為HTS所合成出的茉莉花樂曲之第一樂句的笛聲信號波形，使用7個狀態的HMM模型，圖14則是將圖13的振幅包絡作調整後所得到的波形。觀察波形外圍之振幅包絡曲線，可知調整後的比調整前的變得平滑、自然很多。

5 合成笛聲之評估

關於合成笛聲之效能評估，一種評估方式是客觀量測，把真人吹奏及電腦合成的曲笛聲信號分別算出各音框的頻譜係數，再據以量測對應音框之間的頻譜誤差距離；另一種評估方式是主觀的聽覺測試，由試聽者判斷合成的曲笛聲音的品質。

5.1 客觀距離量測

在客觀量測方面，2.3.1節曾提到音符音長是否細分的兩種作法，當音符音長沒細分時，所訓練出的模型稱為"HMM-DUR1"，而當音符音長有細分為

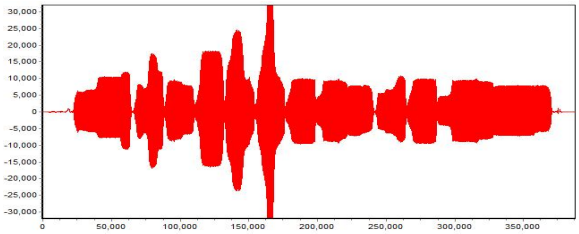

圖 13 未調整振幅包絡的笛聲信號波形

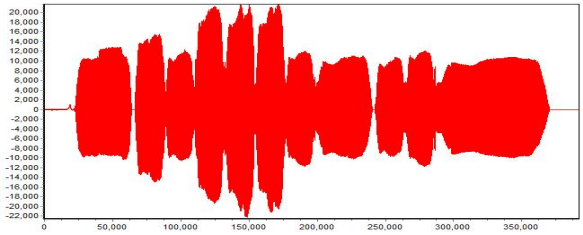

圖 14 調整振幅包絡後的笛聲信號波形

長音與短音時，所訓練得到的模型稱為"HMM-DUR2"；另外，當有作F0取代和弦波式內插的處理，則加上"+F0"縮寫。此外，HMM模型的狀態數，我們分別設定成5個狀態、7個狀態及9個狀態，所以不同狀態數的HMM要分別去量測頻譜誤差。

我們把合成出的及真人吹奏的笛聲樂曲分別拿去分析出一序列音框的MGC係數，然後計算兩序列對應音框的平均誤差距離。距離計算方式是，把合成音音框MGC係數的C_2到C_{34}，拿去和原始笛聲錄音檔中對應音框MGC係數的C_2到C_{34}作幾何距離的計算。在此量測的笛聲樂曲有兩首，一首為慢板的"菊花台"，一首為快板的"鳳陽花鼓"，量測出的平均頻譜誤差距離如表3和4所示。

表 3 樂曲"菊花台"量測出的誤差距離

模型　狀態數	5 個狀態	**7 個狀態**	9 個狀態
HMM-DUR1	0.8708	0.8574	0.8810
HMM-DUR2	**0.8357**	**0.8303**	**0.8507**
HMM-DUR1+F0	0.8655	0.8555	0.8755
HMM-DUR2+F0	**0.8366**	**0.8322**	**0.8491**

表 4 樂曲"鳳陽花鼓"量測出的誤差距離

模型　狀態數	5 個狀態	**7 個狀態**	9 個狀態
HMM-DUR1	0.6686	0.6598	0.6735
HMM-DUR2	**0.6519**	**0.6483**	0.6761
HMM-DUR1+F0	0.6696	0.6609	0.6780
HMM-DUR2+F0	**0.6527**	**0.6489**	**0.6735**

依據表3和4的平均誤差距離值，我們觀察到如下的三個結果：(a)快板樂曲的誤差值會比慢板的誤差值小；(b)除了"鳳陽花鼓"在未取代F0和使用9個狀態HMM模型的情況，其它情況下比較HMM-DUR1和HMM-DUR2，全部都是HMM–DUR2的誤差值較小；(c)比較HMM模型狀態數的影響，可發現快板樂曲和慢板樂曲皆是7個狀態的最好，5個狀

態的次之，9個狀態的最差。

5.2 主觀聽測實驗

由於前一節誤差距離量測的結果得知HMM模型的狀態數設為7時最好，所以我們採用狀態數為7的HMM模型去合成笛聲，不過音符音長有無細分的兩種HMM都拿來作實驗，其縮寫代號就如5.1節裡列出的。

在此聽覺測試是，對合成笛聲的自然度作比較，使用的樂曲為內部樂曲(即有參加模型訓練的樂曲)，即快板的"鳳陽花鼓"與慢板的"菊花台"。我們把三種合成方法所合成出的樂曲兩兩拿去作聽測比較，HTS法就是直接以HTS軟體來合成出樂曲，未再作其它的處理；HTS+F0法是把HTS產生的F0軌跡先作取代和弦波式內插，然後再令HTS去合成出信號；HTS+F0+AE法則是把HTS +F0法合成出的音檔，再拿去作振幅包絡的調整。

我們都請了15位受測者來評分，其中9位是有語音處理的研究經驗，6位則是沒有語音處理的研究經驗，給受測者作聽測的網頁在 http://speech9.csie.ntust.edu.tw/lab/qudi/ 。受測者評分的範圍為1到5分，1分表示非常不自然，5分表示非常自然。進行聽測時，我們提供兩個音檔給受測者聽，當作1分和5分的標準，1分的音檔為HTS直接合成且未做過其它處理的音檔，5分的音檔為原始錄音的曲笛樂曲。作聽測的樂曲分別使用HMM-DUR1和HMM-DUR2兩種模型去合成。聽測後算出的平均評分如表5所示，HMM-DUR2模型得到的分數都比HMM-DUR1的高，由此可看出此項聽測的評分結果和5.1節裡的結果2相符合，並且快板樂曲的評分都是比慢板樂曲的評分高。

表5 曲笛聲音自然度聽測之平均分數

菊花台(HMM-DUR1)	3.47
菊花台(HMM-DUR2)	3.87
鳳陽花鼓(HMM-DUR1)	4.13
鳳陽花鼓(HMM-DUR2)	4.20

6 結論

本論文研製曲笛聲音的合成系統，首先使用HTS軟體去訓練出頻譜HMM模型，再令HTS使用HMM模型去對輸入的樂譜作笛聲合成，在中間步驟把音高軌跡取代為我們計算出的正常音高軌跡，然後令HTS再次合成出信號後，接著我們把信號的振幅包絡調整成較自然的包絡。

在效能評估方面，我們拿合成笛聲與原始錄音笛聲分析出的頻譜係數去作客觀的頻譜誤差距離量測，由量測出的平均誤差距離得知，使用音符音長再細分的HMM模型較好，並且HMM模型使用7個狀態數的誤差距離較小。另外在主觀聽測上，由聽測平均評分得知，修改HTS引擎去取代F0軌跡的方法以及作振幅包絡調整的方法，可以讓合成出的笛聲變得比原始HTS合成的笛聲更為自然很多。

本研究未來可以改進的地方，首先是在錄製笛聲樂曲的準備方面，可以盡量讓長音與短音音符的數量更平衡；在HTS訓練方面，設計不同的問題集，也是值得去嘗試的；另外在振幅包絡的模型方面，可以嘗試使用更好的模型。

參考文獻

[1] 王如江，基於歌聲表情分析與單元選擇之國語歌聲合成研究，碩士論文，國立台灣科技大學資訊工程研究所 2007。

[2] 李振宇、林奇嶽，"使用隱藏式馬可夫模型為基礎建立中文語音合成系統"，ICL Technical Journal, pp. 88-94, 2010.

[3] 黃仕偉，基於小提琴技法之音樂合成，碩士學位論文，成功大學資訊工程學系，2011。

[4] 陳安璿，整合 MIDI 伴奏之歌唱聲合成系統，碩士論文，國立台灣科技大學資訊工程研究所 2004。

[5] 劉冠驛，基於隱藏式馬可夫模型之英文語音合成系統實作，碩士論文，交通大學電信工程系所，2011。

[6] D. Dimitriadis and P. Maragos, "Robust energy demodulation based on continuous models with application to speech recognition", in Proc. of Eurospeech, Geneva, Sept. 2003.

[7] H. Kawahara, I. Masuda-Katsuse and A. de Cheveigne, "Restructuring speech representations using a pitch-adaptive time-frequency smoothing and an instantaneous-frequency based F0 extraction," Speech Communication, vol. 27, pp. 187- 207 , 1999.

[8] A. V. Oppenheim and R. W. Schafter, Descrete -time Signal Processing, second ed., Prentice-Hall, 1999.

[9] L. Rabiner and B. H. Juang, Fundamentals of Speech Recognition, Prentice-Hall, 1993.

[10] K. Shinoda and T. Watanabe, "Acoustic modeling based on the mdl principle for speech recognition," Rhodes, Greece, September 22-25, ISCA, 1997.

[11] SPTK Working Group , Speech Signal Processing Toolkit (SPTK), http://sp-tk.sourceforge.net/.

[12] J. Yamagishi, An Introduction to HMM-based Speech Synthesis, Technical report, Tokyo Institute of Technology, October 2006.

[13] S. Young, G. Evermann, T. Hain, D. Kershaw, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. Woodland, The HTK Book (for HTK version 3.2.1), Cambridge University Engineering Department, 2002.

[14] H. Zen, K. Tokuda, K. Oura, K. Hashimoto, S. Shiota, S. Takaki, J. Yamagishi, T. Toda, T. Nose, S. Sako, A. W. Black, HMM-based Speech Synthesis System (HTS), http://hts.sp.nitech.ac.jp/ .