

# 資工系實務專題研究計畫成果報告

## 《MP3 哼唱辨識系統》

專題編號：100-CSIE-S002

執行期限：99 年 1 學期至 100 年 1 學期

指導教授：楊士萱 教授

專題參與人員：97590324 楊先絜、97590329 莊 璜、97590350 李璧帆

### 一、中文摘要

很多時候我們只記得旋律或是聽到好聽的歌曲，卻不知道歌名時，可藉由匯入歌曲或是哼唱的方式來找尋歌曲，並顯示此歌曲的相關資訊。

本專題核心為音訊辨識。利用 MP3 〈MPEG-1 Audio Layer 3〉解碼過程中可取得的 MDCT 〈Modified Discrete Cosine Transform〉係數，當作音樂特徵值，並在每個 frame 中找能量較高的 MDCT 係數，用來進行資料庫的歌曲與輸入歌曲比對，供使用者找尋歌曲的需求，並播放其歌曲 MV 和顯示專輯資訊，方便使用者不必另外再搜尋。

關鍵詞：MP3 〈MPEG-1 Audio Layer 3〉、MDCT 〈Modified Discrete Cosine Transform〉、哼唱式查詢(Query by Humming)。

### 二、緣由與目的

隨著現在歌曲日益的增多，有太多被我們遺忘的歌跟很多耳熟能詳的歌曲，只記得旋律卻忘記歌名，或者在逛街的時候偶然聽到很喜歡的歌卻不知道歌名時，因為無從搜尋起，結果造成喜歡卻無法再度欣賞的遺憾。為此，本專題將藉由哼唱一小段歌曲的方式，或是將聽到的音樂當場錄下來，接著進行比對來找尋歌曲，進而解決了苦於不知道歌曲名稱的煩惱，並體貼使用者不必再多做個步驟去做搜尋 MV 和專輯相關資訊。

### 三、系統功能與技術說明

本專題的核心方向為 MP3 歌曲辨識，透過「MP3 解碼」時取得的 MDCT 係數做為辨識時的數據。我們參考 ISO 的開放原

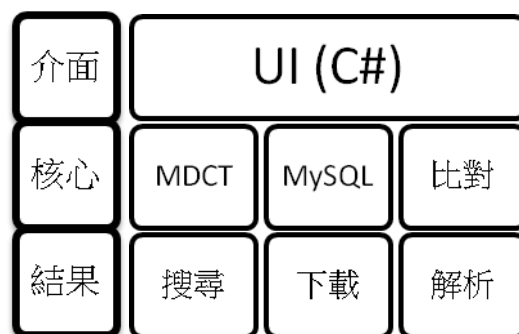
始碼，並整合至 C# 環境使用。

#### 3.1 系統功能

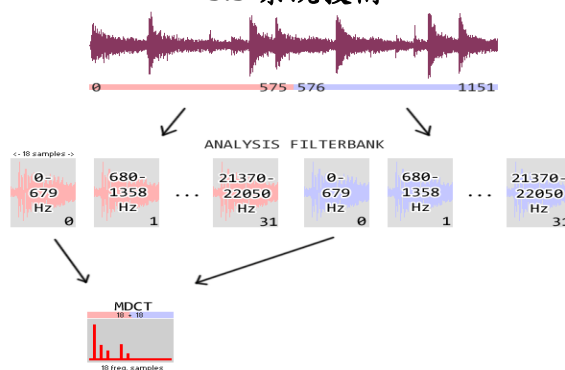
使用者可匯入歌曲，或是即時哼唱歌曲來與系統資料庫裡的歌曲做辨識，辨識完後會顯示相近的歌曲，且使用者可以看歌曲的 MV 和歌詞等專輯資訊並附有轉檔小工具供使用者使用；使用者也可匯入歌曲用來進行 K 歌大賽；且時常被搜尋的歌曲會顯示於熱門排行榜中。

#### 3.2 系統架構

圖一. 系統架構圖



#### 3.3 系統技術



圖二. 取樣頻率為 44.1KHZ 的分析圖

MP3 預設取樣頻率為 44.1KHz，它是依人耳的聽覺範圍 20Hz~22.05KHz 二倍的頻率來取樣。而我們辨識時取大部分演唱者哼唱的頻率範圍 192Hz~958Hz 左右。

MDCT 特徵係數與頻率成線性關係，且半個 frame 分成 576 個等寬的子頻帶，因此可以計算出一個子頻帶的寬度大約為

38.28125Hz。

下表為我們計算後整理的 MDCT 與音符頻率對應表。

表一. 各音譜對應頻率

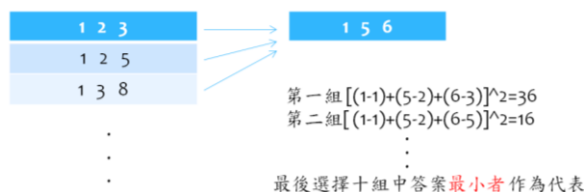
| Mdct<br>係數 | 4  | 5   | 6  | 7  | 8   | 9  | 10  | 11 | 12  | 13  | 14  |
|------------|----|-----|----|----|-----|----|-----|----|-----|-----|-----|
| 音符         | -F | -G# | C  | C# | D#  | F  | G   | A  | A#  | 2C  | 2C# |
| 轉換<br>數值   | 1  | 4   | 8  | 9  | 11  | 13 | 15  | 17 | 18  | 20  | 21  |
| Mdct<br>係數 | 15 | 16  | 17 | 18 | 19  | 20 | 21  | 22 | 23  | 24  |     |
| 音符         | 2D | 2D# | 2E | 2F | 2F# | 2G | 2G# | 2A | 2A# | 3D# |     |
| 轉換<br>數值   | 22 | 23  | 24 | 25 | 26  | 27 | 28  | 29 | 30  | 35  |     |

接著我們將資料庫的歌曲找出每個 frame 中前三高的能量，輸入的歌曲則解析出每個 frame 中前五高的能量，輸入的歌曲找比較多的能量是因為使用者可能有些微偏差音準，如此一來便可以提高比對率。

$C_3^5$  列出 10 種排列組合

使用者(1 2 3 5 8)

資料庫



圖三、比對方式

比對方式如上圖，假設在某個 frame 中，使用者能量為 1、2、3、5、8，資料庫為 1、5、6，使用  $C_3^5$  的方式，將所有的組合計算出值後找出最小值作為代表，最後使用 Sliding Window 計算出每個 frame 的平均值後找出 Window 最小的值作為代表，並與資料庫的所有歌曲比對，前幾首較小值的歌曲即為與使用者輸入較接近的歌。

播放歌曲和相關資訊，分為「下載 HTML」和「解析 HTML」。「下載 HTML」部分，利用 WebClient.DownloadData(String) 下載搜尋後的結果。在此要注意某些網站不支援參雜中文網址，所以要把中文轉為 URL Encoding；「解析 HTML」部分，利用正則表達式〈Regular Expressions〉可以對字串進行比對和取代。

#### 四、目前進度成果

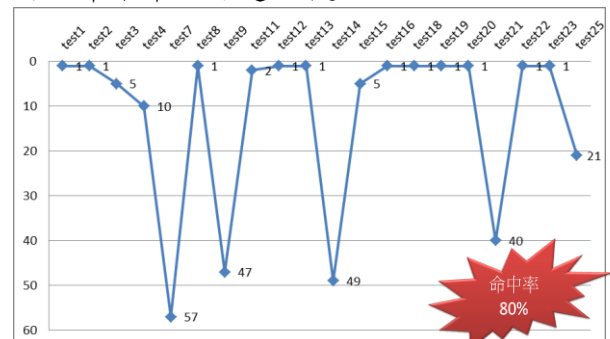
此軟體分為四個功能，第一個功能為

辨識歌曲；第二個功能可將 wav 檔轉成 mp3 檔；第三個功能可進行 K 歌大賽，評語分為四種；第四個功能可看熱門辨識歌曲。



圖四、軟體功能

如下圖是我們找到對的歌曲的排名數據，在資料庫裡有 23 首歌的情況下，命中率達 89%，當資料庫的歌曲增加到 100 首，命中率也可達 80%。



圖五、測試歌排行（資料庫歌曲為 100 首）

#### 五、結論與未來展望

為了改善比對所消耗的時間，我們將原本開關檔案的方式改為動態資料庫連結，雖然已大幅降低比對時間，但仍有改善空間。

由於每個人的 KEY 不同和哼唱的節奏不同，造成分析後的數據差距較大，無法順利找到相近的歌曲，希望藉由調 KEY 和 Dynamic Time Warping 來改善此問題。

#### 六、參考文獻

- [1] MP3 解碼原始碼  
<http://www.mp3-tech.org/>
- [2] Let's build an MP3-decoder!  
<http://blog.bjrn.se/2008/10/lets-build-m-p3-decoder.html>
- [3] 陳毅修、劉志俊，中華大學論文「MP3 音樂哼唱式查詢在嵌入式系統上的實作」

