

中華民國第四十六屆中小學科學展覽會
作品說明書

高中組 生活與應用科學科

第三名

040810

識破你的「蛙」言巧語

學校名稱： 國立新竹女子高級中學

作者： 高二 陳尹安 高二 蘇奕鴿 高二 徐于康 高二 張廖曉蓉	指導老師： 徐以誠 郭美奐
----------------------------------------------	---------------------

關鍵詞：蛙聲辨識、梅爾倒頻譜分析

壹、摘要：

在本篇研究中我們運用現有人類語音處理技術中的能量法與梅爾倒頻譜分析法，發展出了一套利用蛙聲來自動辨識其品種的系統，首先，利用能量法萃取出蛙聲中的音節，然後從音節中計算出梅爾倒頻譜參數 MFCC，利用這些音節之梅爾倒頻譜參數序列建置起台灣地區 32 種蛙類聲音的特徵資料庫。待辨識之蛙聲進入辨識系統後，也可以同樣的方法來處理，取出其各音節的梅爾倒頻譜參數序列後，再透過蛙聲辨識的方法與資料庫比對而得到辨識結果。從程式的實作與辨識能力的測試中得知，我們的方法可以在很短的時間內就獲得辨識的結果，而且可以百分之百地精確辨識出所有的測試蛙聲。

貳、研究動機與目的：

近年來隨著人口的成長及經濟發展，自然生態環境受到極大的破壞，許多青蛙面臨生存的威脅。在都市的喧囂中，幾乎聽不見青蛙純樸的鳴聲，更不用說親眼見到活蹦亂跳的牠們了！在機緣巧合之下，我們來到了某間生態農場，一聲「呱呱」引起了我們的注意，循聲探訪，嘹亮的天籟突然地從草叢竄出。被包圍的我們，震懾於這美好的一刻，疲憊的心靈頓時昇華，這是屬於青蛙的合唱之夜。

除了讚嘆蛙鳴的美妙，我們希望能更進一步的了解牠們。環顧四周，卻不見青蛙的蹤跡，加上我們又是門外漢，無法光憑青蛙叫聲來認識牠們。於是，我們靈機一動，希望發展出一套簡易又快速的程式，能夠以蛙聲來自動判別青蛙的種類，那就太好了。現代社會人手一機，先進的手機也隨著技術的進步而多附有影音多媒體與行動通訊的功能，如果錄音取得想要的蛙鳴聲之後，直接透過通訊網路將聲音檔案傳送至蛙鳴資料庫查詢，經過分析辨識，再把結果傳送回來，這樣一來不但節省時間也十分方便。

參、器材及軟體：

- 一、個人電腦
- 二、Matlab 6.5
- 三、GoldWave 5.06
- 四、Visual Basic 6
- 五、錄音筆

肆、研究過程：

一、語音辨識簡介

利用電腦技術來處理語言問題，是現代科技中一項重要的發展，最終的目標，就是希望人類與機器可以使用自然語言來溝通。語音辨識的研究於 1950 年代開始發展至今，有幾項非常重要的技術成果，例如快速傅利葉轉換 FFT、線性預估編碼 LPC、線性倒頻譜

分析 LPCC、梅爾倒頻譜分析 MFCC 等，這些技術給後來的語音辨識發展，奠定了非常重要的基礎。除了前述的語音分析技術之外，在語音的比對上也發展出三個重要的工具，一是動態時間校正 DTW，二是隱藏式馬可夫模型 HMM，三是類神經網路 ANN，目前隱藏式馬可夫模型 HMM 與類神經網路 ANN 已成為語音辨識的標準工具[2]。雖然陸續有人提出關於蟋蟀、鳥類、青蛙等聲音辨識的文獻報告[4][5][6]，但相較於人類語音辨識技術的蓬勃發展[1][2]，動物聲音辨識的研究就實在是少的可憐，Taylor et al.[6] 利用聲譜圖上的一些波峰高點來辨識澳洲北部地區的 22 種蛙類，根據其研究結果顯示，他們所發展出來的系統有不錯的辨識能力，由於辨識程式的運作是以聲譜圖為基礎，所以會較為耗費時間。

二、檢視「蛙」言巧語

與人類複雜的語音比較起來，蛙類的鳴叫聲顯得單純的多，使用 GoldWave 軟體來觀察蛙類的聲音在時間與頻率上的變化，可以發現青蛙的鳴聲在同一音節內，頻率分佈的範圍較為集中且變化不大（圖 1），所以我們打算先偵測出整個聲音訊號中蛙聲音節的所在，然後以各音節的頻率特徵值來辨識蛙聲。

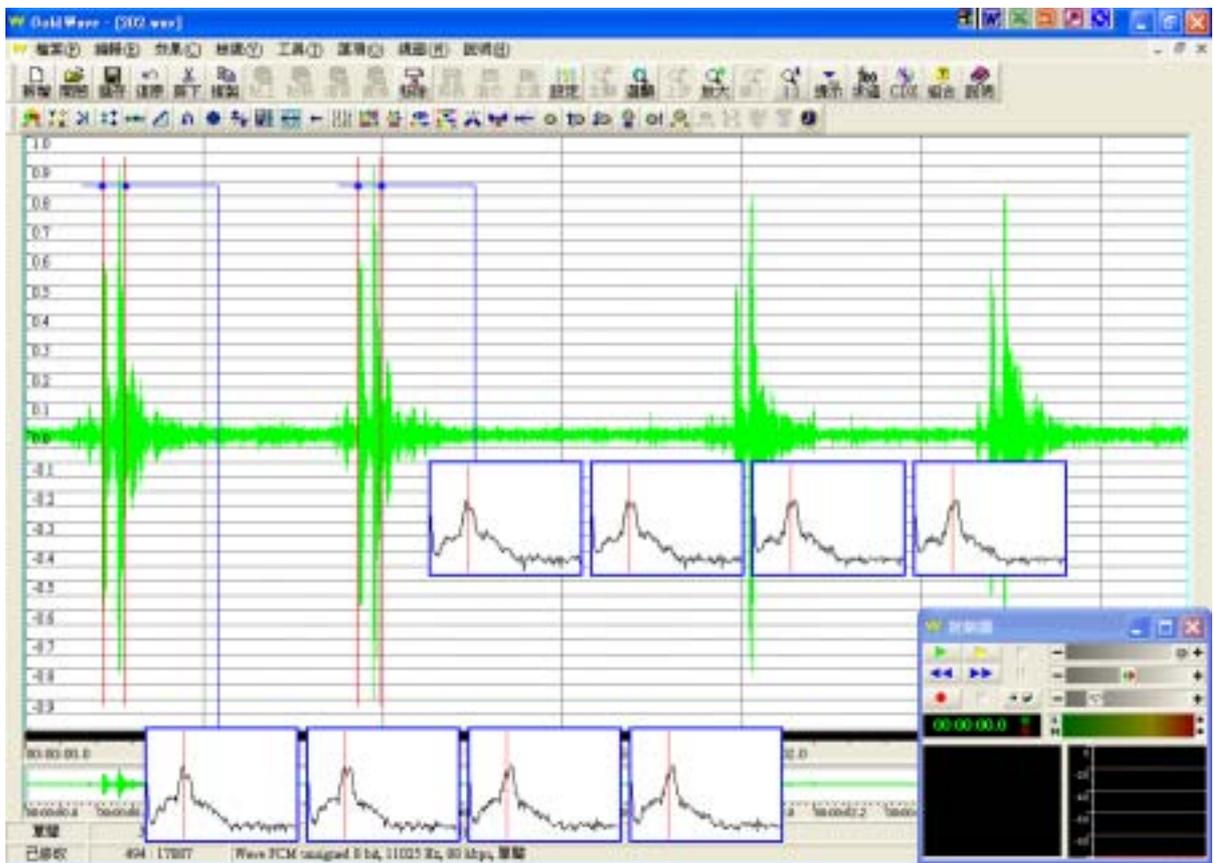


圖 1 蛙聲在同一音節內頻率分佈的範圍較為集中且變化不大

三、蛙聲辨識系統

蛙聲辨識系統主要分成蛙聲特徵資料庫的建立以及辨識蛙聲的方法兩個部分，藉著蛙聲音節的萃取與各音節特徵值的計算建立蛙聲特徵資料庫，待辨識之蛙聲也可以同樣的方法來處理，再透過蛙聲辨識的方法與資料庫比對而得到辨識結果，整個系統可用圖 2 來表示。

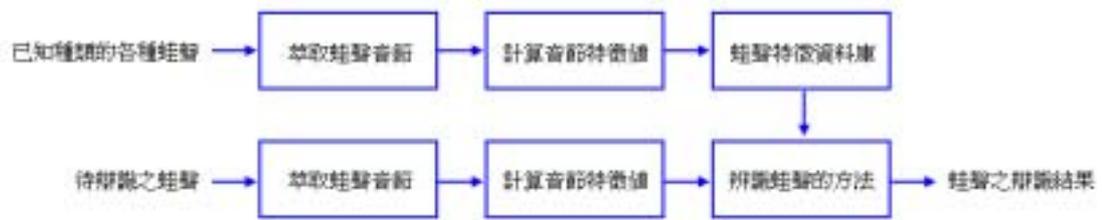


圖 2 蛙聲辨識系統示意圖

四、蛙聲音節的萃取

音節的萃取主要是將整個聲音訊號中，真正屬於蛙聲部分的開始和結束位置標示出來，人類語音處理的技術當中可以採用的方法有很多，主要分成時間域與頻率域兩大類 [1][2]，由於蛙類的鳴叫聲較為單純，不像人類有分氣音與非氣音兩種語音，所以我們採用最簡易的能量法來萃取蛙聲音節。如圖 3 中所示，先繪出聲音訊號的能量曲線，只要音量高於某個門檻值，我們就認定是蛙聲，其他部分就屬於靜音或是雜訊，至於這個門檻值如何決定，需要經過實作測試後方可決定。

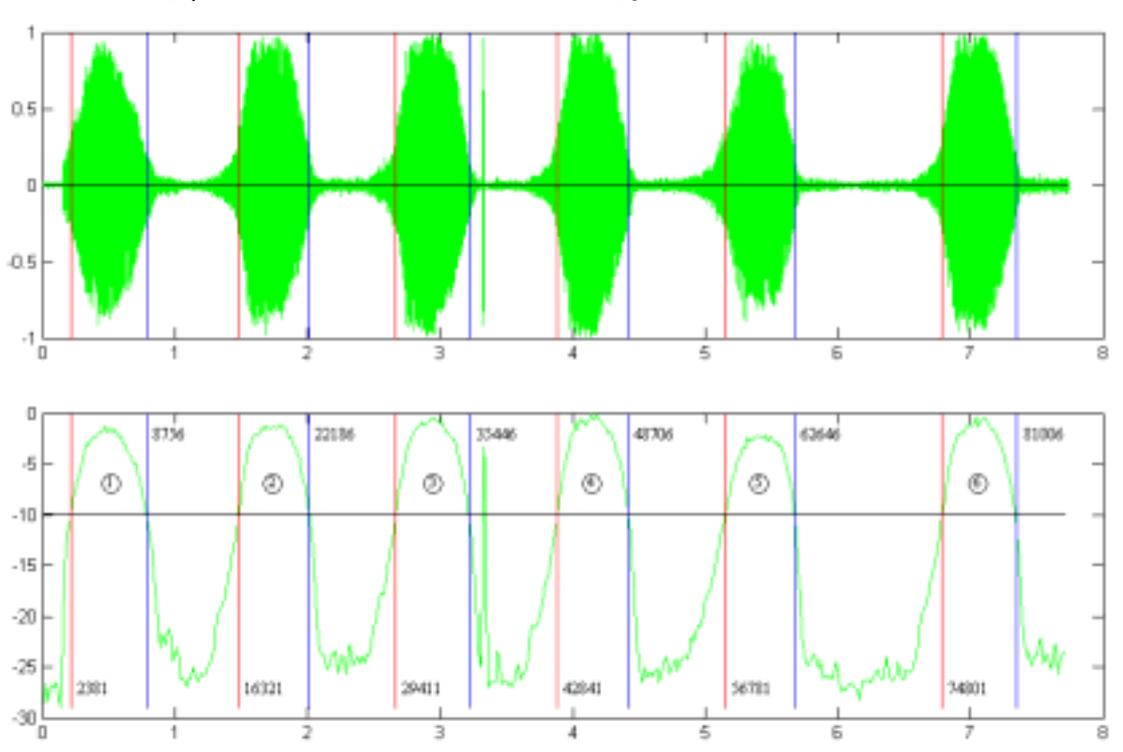


圖 3 利用能量法來萃取聲音訊號中的蛙聲音節

五、音節的特徵值

所謂音節的特徵值就是要找出一組能正確表現該音節特性的參數，而且希望具有鑑別性、能夠很方便地計算兩個音節間的差異以及不易受到外界環境雜訊的干擾，在現有與頻率有關的語音處理技術當中，我們採用了梅爾倒頻譜分析 MFCC 法，由參考資料 [1][2] 知道，梅爾倒頻譜分析 MFCC 法不僅具有前述的優點，而且能將音節的頻譜資訊以符合人耳聽覺特性的方式表現出來。

以梅爾倒頻譜分析 MFCC 法來處理音節訊號，需經過下列一連串的处理與計算後方可獲得一組該音節的梅爾倒頻譜參數作為其特徵值 [1]，整個處理與計算的流程示意圖如圖 4 所示。

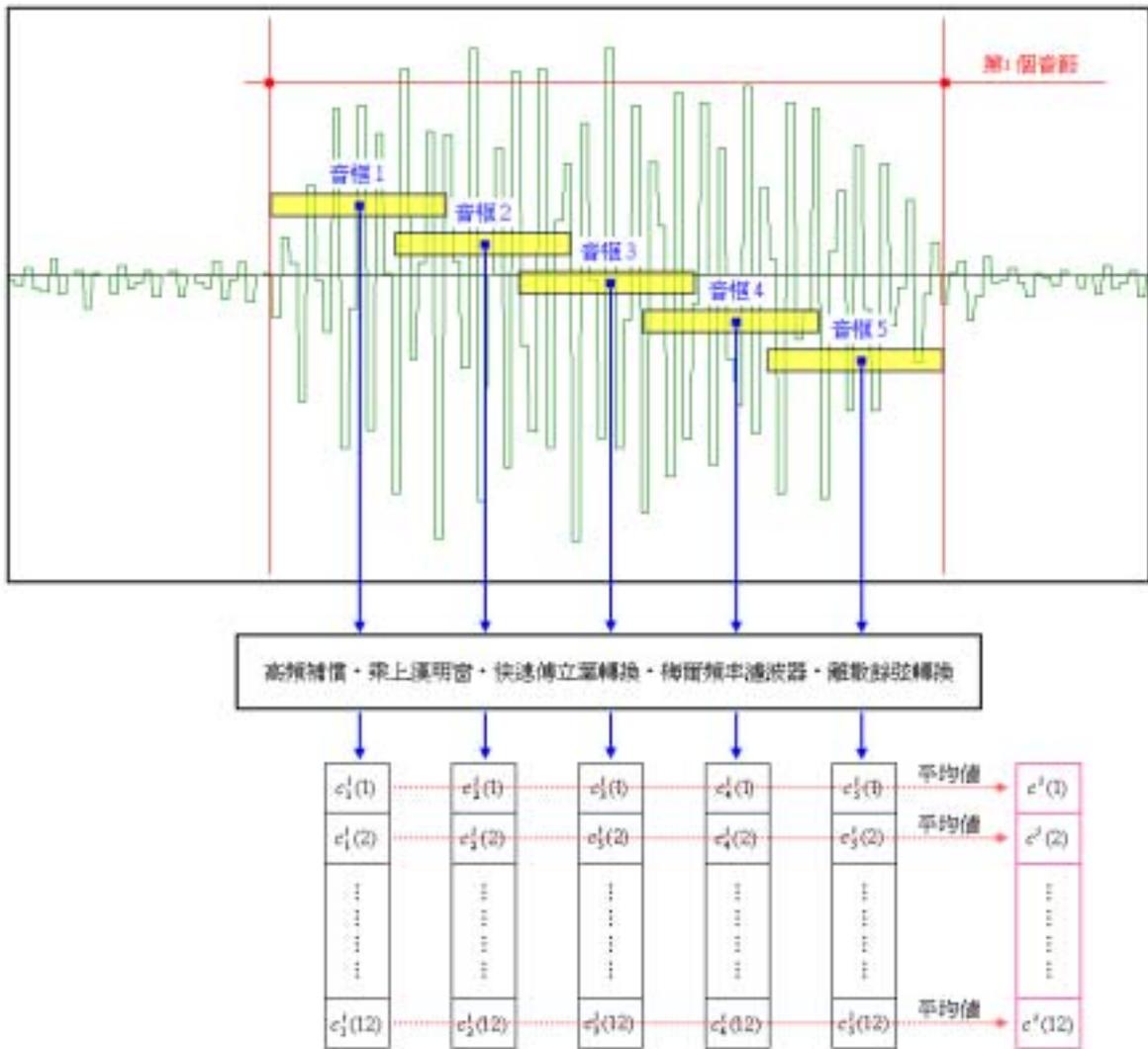


圖 4 計算蛙聲音節的梅爾倒頻譜參數 MFCC 流程示意圖

音框分割

聲音訊號是由許許多多的取樣點所構成，取樣頻率為 11025Hz 的聲音訊號，表示每秒鐘有 11025 個取樣點，若是直接以取樣點來做處理，則資料量太於過龐大，通常都會將其分割成一個個的音框，音框就成為聲音訊號處理的最小單位。將音節中每 256 個音訊取樣點為一個音框，為了避免相鄰的兩音框變化太大，所以音框與音框之間重疊 86 個取樣點，如此可以避免兩相鄰音框之間變化太過劇烈。

高頻補償

聲音訊號自發出後到錄音設備接收，會有高頻的損失，為了還原聲音訊號而做高頻補償處理，可以下列式子來表示，其中 s 表示音框中的原始取樣點訊號； s' 表示經高頻補償處理後的訊號； $n = 0 \sim 255$ 。

$$s'(0) = s(0)$$

$$s'(n) = s(n) - 0.95 \times s(n-1)$$

乘上漢明窗

將每一個音框乘上漢明窗，其目的是消除原本各個音框的兩端可能會造成的訊號不

連續性，音框乘上漢明窗可用下列式子表示，其中 s' 表示音框中經過高頻補償取樣訊號； $h(n)$ 表示漢明窗； s'' 表示乘上漢明窗後的訊號； $n = 0 \sim 255$ ； $N = 256$ 。

$$s''(n) = s'(n) \times h(n)$$

$$h(n) = 0.54 - 0.46 \cos(2n\pi / N - 1)$$

快速傅立葉轉換

快速傅立葉轉換可將音框由時間域的訊號轉換為頻率域的頻譜，其轉換的方式如下式所示，其中 s'' 表示音框乘上漢明窗後的取樣點訊號； S 為頻譜中各頻率的成分； $k = 0 \sim 255$ ； $N = 256$ 。

$$S(k) = \sum_{n=0}^{N-1} s''(n) \times e^{-2kn\pi/N}$$

梅爾頻率濾波器

將音框的頻譜資料通過一組以梅爾量度劃分頻帶的三角帶通濾波器，將屬於同一個頻帶中的頻率成分合在一起看成該頻帶的能量強度，算式如下，其中 $B_m(k)$ 表示第 m 個頻帶的三角帶通濾波器； f_m 為第 m 個頻帶的中心頻率， f_{m-1} 與 f_{m+1} 就是相鄰的前後兩個頻帶之中心頻率； S 為音框頻譜中各頻率的成分； $X(m)$ 為通過第 m 個頻帶三角帶通濾波器的能量強度； M 是全部的頻帶數目； $m = 1 \sim M$ 。

$$X(m) = \log\left\{\sum_{k=f_{m-1}}^{f_{m+1}} |S(k)|^2 \times B_m(k)\right\}$$

離散餘弦轉換

將上述通過梅爾頻率濾波器的 M 頻帶能量作離散餘弦轉換，就可以得到該音框的梅爾倒頻譜參數 $c(1)$ 、 \dots 、 $c(L)$ ，轉換公式如下，在一般的情形下，用前 12 個參數就已足夠 ($L = 1 \sim 12$)。

$$c(L) = \frac{1}{M} \times \sum_{m=1}^M X(m) \cos(L(m-0.5)\pi/12)$$

計算音節的梅爾倒頻譜參數

當音節中每個音框的梅爾倒頻譜參數求得後，我們將每個音框的梅爾倒頻譜參數相同分量之平均值作為音節的梅爾倒頻譜參數，音節的 12 個梅爾倒頻譜參數計算方式如下，其中 c^i 表示蛙聲中第 i 個音節的梅爾倒頻譜參數； c_j^i 表示蛙聲中第 i 個音節的第 j 個音框的梅爾倒頻譜參數； Y^i 表示第 i 蛙聲音節中音框的總數； $L = 1 \sim 12$ 。

$$c^i(L) = \frac{1}{Y^i} \sum_{j=1}^{Y^i} c_j^i(L)$$

六、蛙聲特徵資料庫的建立

當蛙聲經過音節的萃取以及計算出每一個音節的梅爾倒頻譜參數 MFCC 後，就將這些參數當作該蛙聲的特徵資料，當我們完成台灣地區 32 種蛙類[3]的特徵資料計算後，就完成了台灣地區的蛙聲特徵資料庫，整個資料庫的內容大致如圖 5 所示。

蛙類名稱	各音節梅爾倒頻譜參數 MFCC			
種類 1	$c^1(1) \dots c^1(12)$	$c^2(1) \dots c^2(12)$	$c^3(1) \dots c^3(12)$	
種類 2	$c^1(1) \dots c^1(12)$	$c^2(1) \dots c^2(12)$	$c^3(1) \dots c^3(12)$	
種類 3	$c^1(1) \dots c^1(12)$	$c^2(1) \dots c^2(12)$	$c^3(1) \dots c^3(12)$	
⋮	⋮	⋮	⋮	⋮
種類 32	$c^1(1) \dots c^1(12)$	$c^2(1) \dots c^2(12)$	$c^3(1) \dots c^3(12)$	

圖 5 台灣地區的蛙聲特徵資料庫內容

七、辨識蛙聲方法

待辨識的蛙聲進入系統後，將其各音節的梅爾倒頻譜參數 MFCC 序列與資料庫中的每一種類蛙聲作比對，看看待辨識的蛙聲與哪一種類差異度最小，辨識系統就會判定這是什麼種類的蛙聲。假設待辨識的蛙聲共有 m 個音節，其各音節的梅爾倒頻譜參數序列為

$$[c^1(1)\cdots c^1(12)]、[c^2(1)\cdots c^2(12)]、\dots、[c^m(1)\cdots c^m(12)]$$

蛙聲特徵資料庫中，種類 S 的蛙聲有 n 個音節，其各音節的梅爾倒頻譜參數序列為

$$[\text{種類}_S c^1(1)\cdots \text{種類}_S c^1(12)]、[\text{種類}_S c^2(1)\cdots \text{種類}_S c^2(12)]、\dots、[\text{種類}_S c^n(1)\cdots \text{種類}_S c^n(12)]$$

則待辨識的蛙聲與特徵資料庫中種類 S 的蛙聲間之差異度定義如下

$$Difference_{\text{待辨識}}^{\text{種類}_S} = \text{Minimum}_{j=0\sim(n-m)} \left\{ \frac{1}{m} \sum_{i=1}^m \sum_{L=1}^{12} |c^i(L) - \text{種類}_S c^{j+i}(L)| \right\} \quad n \geq m$$

舉例說明可能更為清楚，假設待辨識的蛙聲有 3 個音節，資料庫中種類 S 的蛙聲有 5 個音節，其個別的音節梅爾倒頻譜參數序列如圖 6 所示，哪麼待辨識的蛙聲與特徵資料庫中種類 S 的蛙聲間之差異度計算方式如下：

計算出待辨識蛙聲與種類 S 蛙聲之 1、2、3 音節間的差異度，然後計算出差異度的平均值為 53.21。

計算出待辨識蛙聲與種類 S 蛙聲之 2、3、4 音節間的差異度平均值為 49.59。

計算出待辨識蛙聲與種類 S 蛙聲之 3、4、5 音節間的差異度平均值為 43.54。

待辨識的蛙聲與特徵資料庫中種類 S 的蛙聲間之差異度為平均值中最小者 43.54。



圖 6 待辨識的蛙聲與特徵資料庫中種類 S 的蛙聲間之差異度計算

伍、研究結果：

一、蛙聲的蒐集

要自行到野外蒐錄台灣地區 32 種蛙類的蛙聲，真的是一件很困難的事情，就算蒐錄到了蛙聲，也很難判別是哪一種蛙類的鳴聲，經過我們四處地求援，幸經各大專院校教授學者及生態農場負責人等的大力幫忙，總算所有的蛙聲都蒐集齊全（每種類有 1 至 3 個原始蛙聲音訊檔），我們以數位音訊編輯軟體 GoldWave 將原始蛙聲音訊檔做切割與格式轉換，每一種蛙類有 3 個資料庫音訊檔（10 至 20 秒），2 個測試音訊檔（2 至 4 秒），取樣頻率均為 11025Hz、8bits、單音，檔案格式為 wav。

二、實作軟體的選擇

Matlab 軟體提供了信號處理工具箱 (Signal Processing Toolbox)，用來處理數位音訊檔是最方便不過了，不過在使用者介面與資料庫連結方面則較為薄弱，所以我們打算整合 Visual Basic 6 與 Matlab 軟體[1][7]，以 Visual Basic 來設計使用者介面及連結資料庫，呼叫 Matlab 來處理複雜的聲音訊號。

三、萃取出蛙聲中的音節

我們使用 Matlab 程式來萃取蛙聲中的音節，資料庫音訊檔萃取出 11 至 20 個音節，測試音訊檔萃取出 1 至 6 個音節。音節偵測的方法中，以能量來判定是最為簡潔的作法[1]，由於一般蛙聲音訊檔中，主角 (蛙類) 的聲音通常是比較大聲的部分，所以音量的門檻由高能量值開始萃取，然後逐步往下降，直到萃取出我們需要的音節數目為止，萃取出來的每個音節取樣點起迄位置需記錄下來供後續特徵值 MFCC 計算使用。此外，我們的程式可將時間過短的音節排除 (小於 5 個音框約 0.085 秒)，因為太短的音節可能只是雜訊或是錄音器材所造成，將其排除是很合理的推測。

四、音節特徵值 MFCC 的計算

知道每個音節取樣點的起迄位置後，經過音框分割、高頻補償、乘上漢明窗、快速傅立葉轉換、梅爾頻率濾波器等處理程序，就可以計算出蛙聲中各音節的梅爾倒頻譜參數 MFCC 序列，我們也是利用 Matlab 程式來處理與計算[1]，並將計算出來的蛙聲特徵值傳回 Visual Basic 6 中。

五、建立蛙聲特徵值資料庫

我們 Visual Basic 6 撰寫蛙聲特徵值資料庫的建立程式，透過呼叫 Matlab 程式，以批次的方式依序處理 96 個資料庫音訊檔，並將各蛙聲音訊檔的音節梅爾倒頻譜參數 MFCC 序列存入 Access 資料庫中。

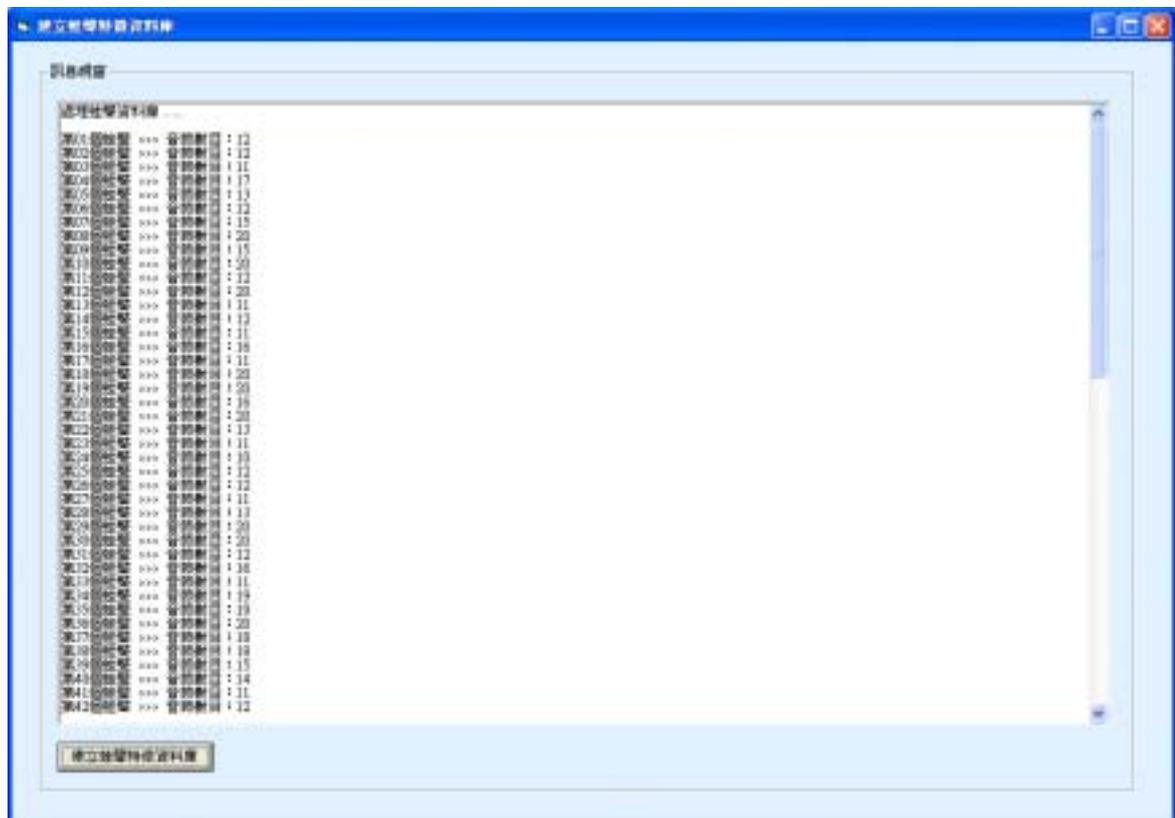


圖 7 以批次處理的方式建立蛙聲特徵值資料庫

六、辨識能力的測試

為了加速辨識能力測試的進行，可先計算出 64 個測試音訊檔的音節梅爾倒頻譜參數 MFCC 序列並存入 Access 資料庫中，然後針對每一個測試音訊與全部的 96 個資料庫音訊作差異度計算，經過差異度的排序後可以知道資料庫音訊中與測試音訊同種類蛙聲的排名，可由程式的實際測試中發現，我們的方法可以精確的辨識出所有的 64 個測試音訊。

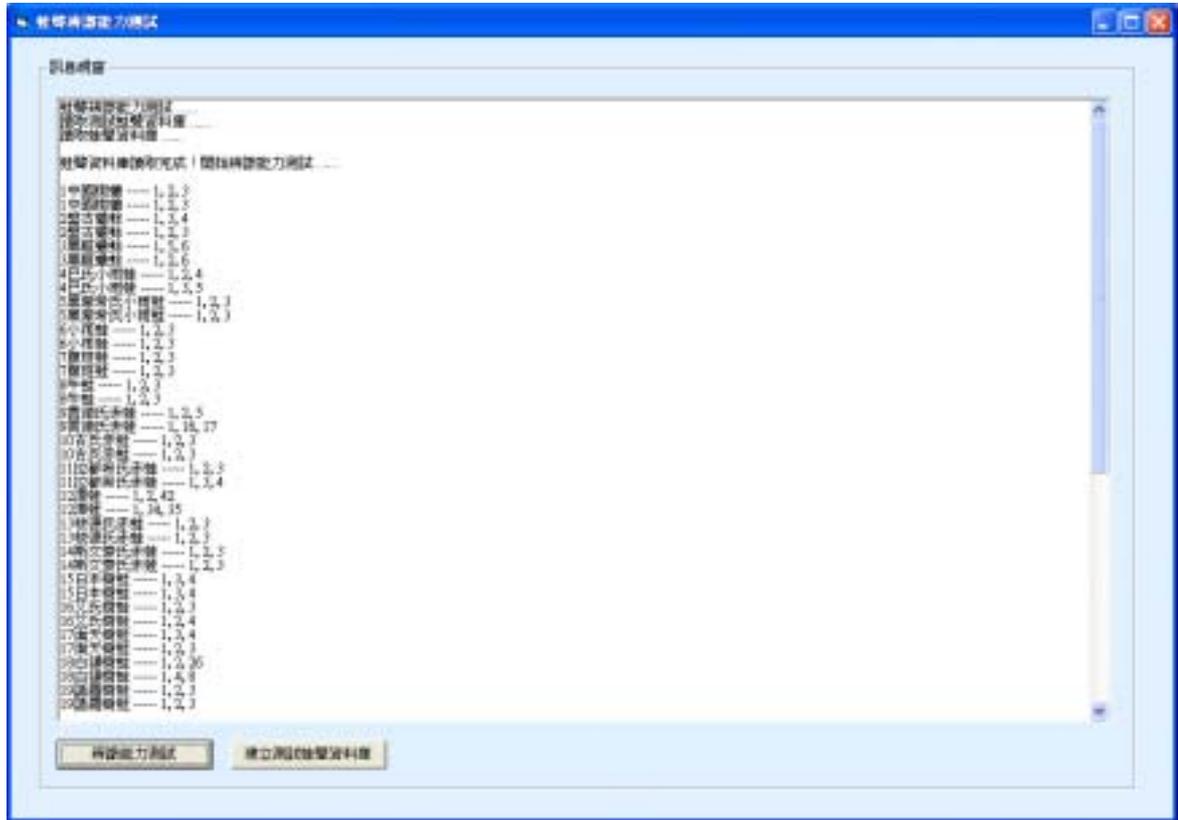


圖 8 辨識能力實作測試發現我們的方法可以辨識出所有測試音訊

七、蛙聲辨識系統

我們整合了 Visual Basic 6 與 Matlab 來設計蛙聲辨識系統，以 Visual Basic 6 來完成資料庫連結以及使用者介面，複雜的蛙聲訊號處理則交給 Matlab 處理，整個系統分為下列四個步驟來完成蛙聲的辨識工作，系統的執行畫面如圖 9 所示。

第一步

由使用者輸入欲辨識的蛙聲，對於查詢的蛙聲有兩項要求，第一，檔案格式為 wav；第二，檔案屬性為 11025Hz、8bits、單音。

第二步

偵測出查詢蛙聲的音節數目，以及起訖取樣點位置。

第三步

計算出查詢蛙聲各音節的梅爾倒頻譜參數 MFCC。

第四步

進行蛙聲辨識比對，比對完成後，依序輸出與查詢蛙聲相似度最高的前 4 種不同種蛙類當做查詢辨識結果。



圖 9 蛙聲辨識系統

陸、討論：

利用電腦來進行蛙聲辨識的工作，基本的要求有兩個：第一、要有快速的辨識速率，太慢易使查詢者失去耐心；第二、辨識的準確度要高，縱使無法達到百分之百，也要在可以接受的範圍內。一般而言，CPU 的運算速度、記憶體的多寡、蛙聲特徵值的多寡及資料庫中蛙聲的總量等，都會影響到系統辨識的效能。為了要提高辨識精確度，資料庫中同種類的蛙聲要多一點才好，如此才能減少同種蛙類間個體差異所造成的誤判，相對地，更形龐大的資料庫則會加長系統辨識時的反應時間。因此，如何同時兼顧兩者是非常重要的課題。

錄音環境的好壞也會影響辨識的結果，目前在多種蛙類同時鳴叫的情況下以及週遭聲音大於蛙類聲音時，我們系統的辨識能力會較差，另外，聲音檔的格式不是 wav 時，系統也是無法運算的。我們希望日後能改善前述的情況，讓使用上的限制減少。

柒、結論：

語音辨識近來是蠻熱門的研究主題，但大多是以人聲資料庫為主，針對動物者更是少之又少。我們藉由能量法與梅爾倒頻譜分析來建立特徵資料庫以及蛙聲比對。從實驗的測試結果得知，這個方法的辨識效能極高，系統反應也能在很短的時間完成，證實我們的策略確實有效。

事實上，並非所有人的手機都有錄音功能，若能更進一步製造一種類似 PDA，可錄音的辨識儀器，在野外錄製聲音後直接進行分析，如此一來又可省下資料傳輸的時間，實用性與便利性也大為提高。

捌、參考資料：

- 一、張智星著，“MATLAB 程式設計與應用，” 清蔚科技, 2000。
- 二、王小川著，“語音訊號處理，” 全華科技, 2004。
- 三、楊懿如著，“賞蛙呱呱叫，” 聯經出版公司, 2003。
- 四、A. L. McIlraith, and H. C. Card, “Birdsong Recognition Using Backpropagation and Multivariate Statistics,” IEEE Trans. on Signal Processing, Vol. 45, No. 11, pp. 2740-2748, 1997.。
- 五、E. D. Chesmore, “ Application of Time Domain Signal Coding and Artificial Neural Networks to Passive Acoustical Identification of Animals, ” Applied Acoustics, Vol. 62, No.12, pp. 1359-1374, 2001.。
- 六、A. Taylor, G. Grigg, G. Watson, and H. McCallum, “Monitoring Frog Communities : An Application of Machine Learning,” Proceedings of Eighth Innovative Applications of Artificial Intelligence Conference, pp. 1564-1569, 1996.。
- 七、洪維恩著，“Matlab7 程式設計，” 旗標出版股份有限公司, 2005。

玖、附錄：

一、台灣地區的 32 種蛙類



01
中國樹蟾



02
盤古蟾蜍



03
黑眶蟾蜍



04
巴氏小雨蛙



05 黑蒙
希氏小雨蛙



06
小雨蛙



07
腹斑蛙



08
牛蛙



09
貢德氏赤蛙



10
古氏赤蛙



11
拉都希氏赤蛙



12
澤蛙



13
梭德氏赤蛙



14
斯文豪氏赤蛙



15
日本樹蛙



16
艾氏樹蛙



17
面天樹蛙



18
白領樹蛙



19
諸羅樹蛙



20
橙腹樹蛙



21
莫氏樹蛙



22
翡翠樹蛙



23
台北樹蛙



24 史丹
吉氏小雨蛙



25
長腳赤蛙



26
金線蛙



27
豎琴蛙



28
虎皮蛙



29
台北赤蛙



30
褐樹蛙



31
花狹口蛙

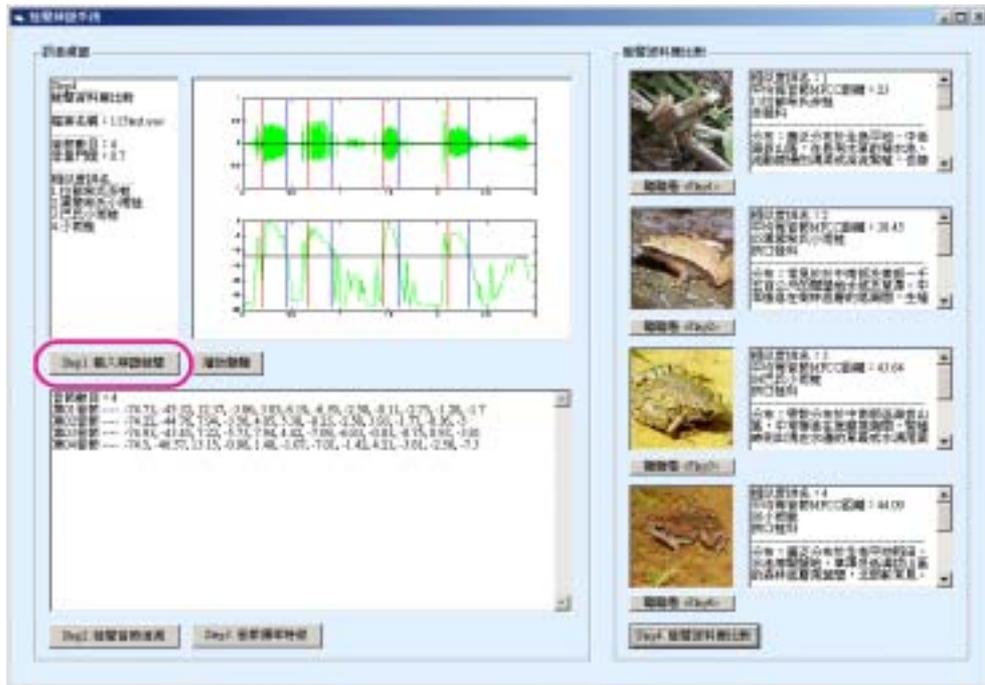


32
海蛙

二、蛙聲辨識系統 - 宣告部分

```
Dim MatLab As Object ' 宣告使用 Matlab 物件
Const MaxTestsyllablenum = 6
Const Maxsyllablenum = 20
Const MFCCnum = 12
Const Maxfrogcallnum = 96
Const Frogsspecies = 32
Const SND_ASYNC = &H1 ' 同步播放
Private Type Frog_Call_type
    speciesid As Integer
    species As String * 16
    family As String * 10
    filename As String * 7
    syllablenum As Integer
    syllableMFCC(Maxsyllablenum - 1, MFCCnum - 1) As Double
End Type
Private Type Match_type
    id As Integer
    speciesid As Integer
    distance As Double
End Type
Private Type Frog_type
    speciesid As Integer
    species As String * 16
    family As String * 10
    JPGfilename As String
    content As String
End Type
Dim FrogCall(Maxfrogcallnum - 1) As Frog_Call_type
Dim Match(Maxfrogcallnum - 1) As Match_type
Dim Inputfile As String
Dim syllableMFCC(MaxTestsyllablenum - 1, MFCCnum - 1) As Double
Dim tmpMFCC() As Double
Dim syllable_num(1) As Double
Dim tmp_num() As Double
Dim syllable_begin_end(MaxTestsyllablenum - 1, 1) As Double
Dim tmp_begin_end() As Double
Dim syllablenum As Integer
Dim thresholdCoef As Double
Dim Frog(Frogsspecies - 1) As Frog_type
Dim WAVEfilename(3) As String
Dim MyDB As New ADODB.Connection ' 資料庫宣告
Dim MyRec As New ADODB.Recordset ' 表單宣告
Private Declare Function sndPlaySound Lib "winmm.dll" Alias "sndPlaySoundA" (ByVal IpszSoundName As String, ByVal uFlags As Long) As Long
```

三、蛙聲辨識系統 - 輸入辨識蛙聲



```

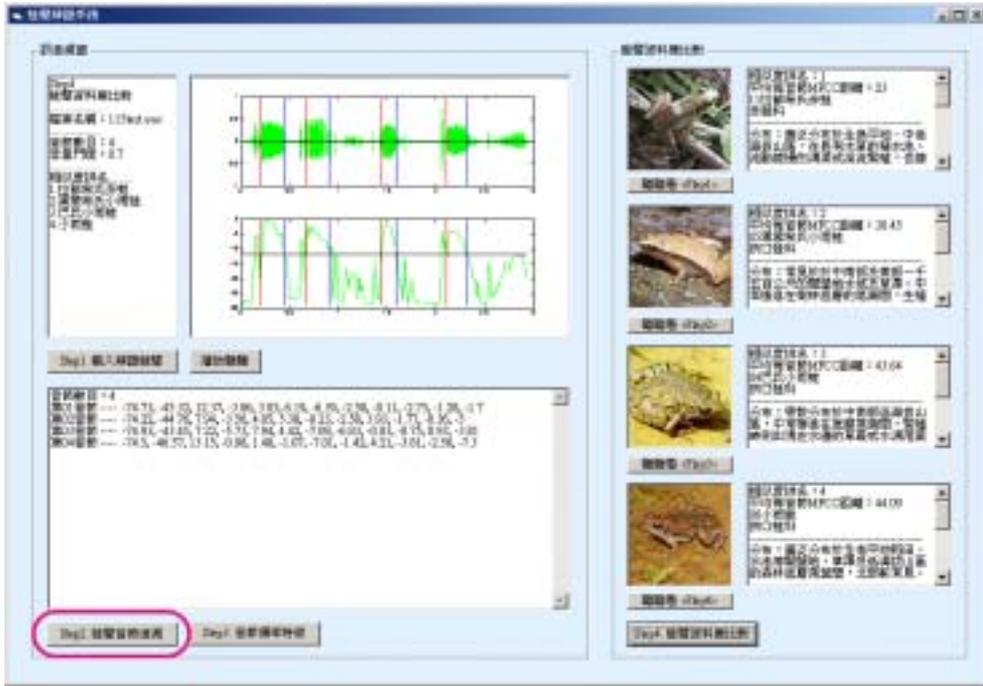
Private Sub btn_InputFrogCall() ' 輸入辨識蛙聲並顯示波形
    Set MatLab = CreateObject("MatLab.Application") ' 建立 Matlab 物件準備呼叫 Matlab
    MatLab.Visible = False
    CommonDialog1.ShowOpen ' 開啟輸入檔案對話窗
    Inputfile = CommonDialog1.filename ' 讀進輸入檔之檔名
    MatLab.Execute ("clear;") ' 清除 Matlab 中變數
    Call MatLab.PutCharArray("filename", "base", Inputfile) ' 將欲辨識之蛙聲檔案名稱傳給 Matlab
    MatLab.Execute ("Recognition_show_wave") ' 執行顯示聲音波形的 Matlab 程式
    MatLab.Execute ("print -dmeta; delete(gcf);") ' 將 Matlab 顯示的圖形放入剪貼簿
    If Clipboard.GetFormat(3) Then
        Picture1.Picture = Clipboard.GetData() ' 將剪貼簿的圖形放入 Picture1
    End If
    Call MatLab.Quit ' 結束 Matlab 物件
End Sub
    
```

四、蛙聲辨識系統 - 聲音播放



```
Private Sub btn_Play() ' 播放蛙聲檔
    sndPlaySound Inputfile, SND_ASYNC
End Sub
```

五、蛙聲辨識系統 - 蛙聲音節偵測



```

Private Sub btn_DetectSyllable() ' 蛙聲音節偵測與音節起訖點顯示
    Set MatLab = CreateObject("MatLab.Application") ' 建立 Matlab 物件準備呼叫 Matlab
    MatLab.Visible = False
    MatLab.Execute ("clear;") ' 清除 Matlab 中變數
    Call MatLab.PutCharArray("filename", "base", Inputfile) ' 將欲辨識之蛙聲檔案名稱傳給 Matlab
    MatLab.Execute ("Recognition_detect_syllable;") ' 執行音節偵測的 Matlab 程式
    Call MatLab.GetFullMatrix("syllable_begin_end", "base", syllable_begin_end, tmp_begin_end) ' 將音節起訖點傳回來 syllable_begin_end
    Call MatLab.GetFullMatrix("Syllable_num", "base", syllable_num, tmp_num) ' 將音節數目傳回來 Syllable_num
    Call MatLab.Quit ' 結束 Matlab 物件
    syllablenum = syllable_num(0)
    thresholdCoef = syllable_num(1)
    MatLab.Execute ("print -dmeta; delete(gcf);") ' 將 Matlab 顯示的圖形放入剪貼簿
    If Clipboard.GetFormat(3) Then
        Picture1.Picture = Clipboard.GetData() ' 將剪貼簿的圖形放入 Picture1
    End If
End Sub
End Sub
    
```

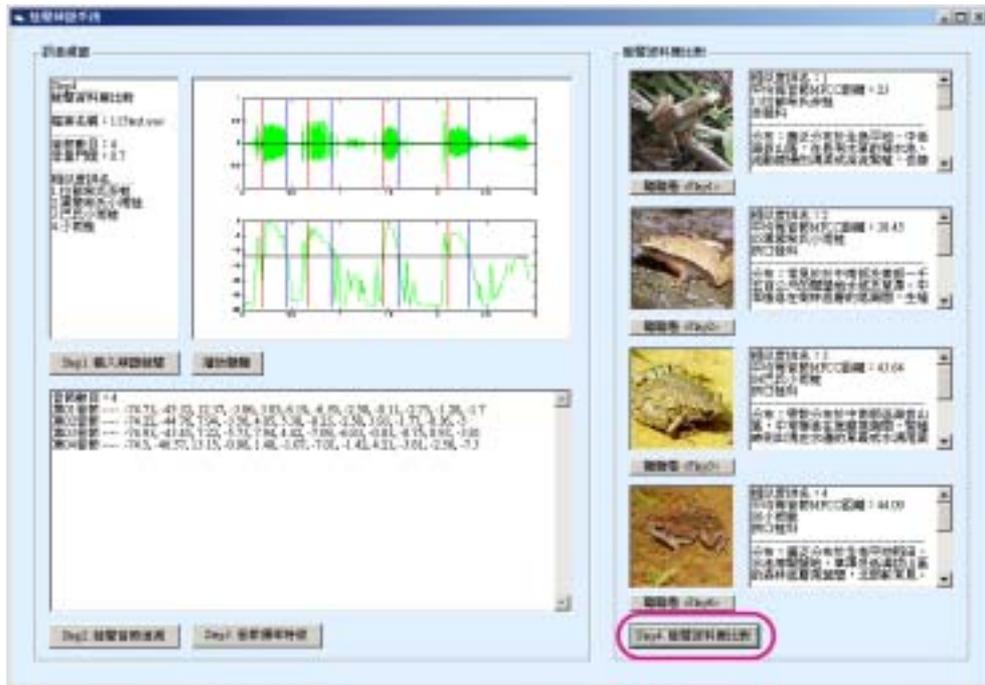
六、蛙聲辨識系統 - 音節梅爾倒頻譜參數 MFCC 計算



```

Private Sub btn_SyllableMFCC() ' 音節梅爾倒頻譜參數 MFCC 計算
    Set MatLab = CreateObject("MatLab.Application") ' 建立 Matlab 物件準備呼叫 Matlab
    MatLab.Visible = False
    MatLab.Execute ("clear;") ' 清除 Matlab 中變數
    Call MatLab.PutCharArray("filename", "base", Inputfile) ' 將欲辨識之蛙聲檔案名稱傳給 Matlab
    MatLab.Execute ("Recognition_syllable_MFCC;") ' 執行音節 MFCC 計算的 Matlab 程式
    Call MatLab.GetFullMatrix("coef", "base", syllableMFCC, tmpMFCC) ' 將音節 MFCC 傳回來 coef
    Call MatLab.GetFullMatrix("Syllable_num", "base", syllable_num, tmp_num) ' 將音節數目傳回來 Syllable_num
    Call MatLab.Quit ' 結束 Matlab 物件
    syllalenum = syllable_num(0)
End Sub
    
```

七、蛙聲辨識系統 - 蛙聲資料庫比對



Private Sub btn_Recognition()

Const RecognitionFrog = 4 ' 輸出相似度最高的前 4 種蛙類

Dim i, j, k, currentCall, currentFrog, SearchFrog, OutputFrog As Integer

Dim currentField As String * 5

Dim tmpDistance, minDistance As Double

Dim tmpMatch As Match_type

Dim RecognitionMatch(RecognitionFrog - 1) As Integer

Dim OutPutOver, Selected, YesInOutput As Boolean

Dim tmpwavfile As String

MyDB.Open "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=D:\46 蛙聲辨識\FrogCall.mdb;Persist Security Info=False" ' 開啟資料庫

MyRec.Open "Frequency", MyDB ' 開啟資料庫中的表單 Frequency

currentCall = 0

Do Until MyRec.EOF ' 將蛙聲特徵資料庫讀入 FrogCall 陣列

FrogCall(currentCall).speciesid = MyRec.Fields("speciesid").Value

FrogCall(currentCall).species = MyRec.Fields("species").Value

FrogCall(currentCall).family = MyRec.Fields("family").Value

FrogCall(currentCall).filename = MyRec.Fields("filename").Value

FrogCall(currentCall).syllablenum = MyRec.Fields("syllablenum").Value

For i = 0 To Maxsyllablenum - 1

For j = 0 To MFCCnum - 1

currentField = "f" & Format(i + 1, "00") & Format(j + 1, "00")

FrogCall(currentCall).syllableMFCC(i, j) = MyRec.Fields(currentField).Value

Next

Next

currentCall = currentCall + 1

MyRec.MoveNext

Loop

MyRec.Close ' 關閉資料庫表單

MyDB.Close ' 關閉資料庫

For i = 0 To Maxfrogcallnum - 1 ' 蛙聲差異度陣列初始化

Match(i).id = i + 1

Match(i).speciesid = FrogCall(i).speciesid

Match(i).distance = 0

Next

For currentCall = 0 To Maxfrogcallnum - 1 ' 計算查詢蛙聲與所有資料庫蛙聲間的差異度

For k = 0 To FrogCall(currentCall).syllablenum - syllablenum

tmpDistance = 0

For i = 0 To syllablenum - 1

For j = 0 To MFCCnum - 1

tmpDistance = tmpDistance + Math.Abs(syllableMFCC(i, j) - FrogCall(currentCall).syllableMFCC(k, j))

Next

Next

Next

```

    tmpDistance = tmpDistance / syllablenum
    If k = 0 Then
        minDistance = tmpDistance
    Else
        If tmpDistance < minDistance Then
            minDistance = tmpDistance
        End If
    End If
Next
Match(currentCall).distance = minDistance
Next
For i = 0 To Maxfrogcallnum - 2 ' 依照差異度由小至大排序
    For j = 0 To Maxfrogcallnum - 2 - i
        If Match(j).distance > Match(j + 1).distance Then
            tmpMatch = Match(j)
            Match(j) = Match(j + 1)
            Match(j + 1) = tmpMatch
        End If
    Next
Next
OutputFrog = 0 ' 輸出相似度最高的前 4 種蛙類
OutPutOver = False
Do
    Selected = False
    SearchFrog = 0
    Do
        YesInOutput = False
        If OutputFrog > 0 Then
            For i = 0 To OutputFrog - 1
                If Match(SearchFrog).speciesid = Match(RecognitionMatch(i)).speciesid Then
                    YesInOutput = True
                End If
            Next
        End If
        If YesInOutput = False Then
            Selected = True
        Else
            SearchFrog = SearchFrog + 1
        End If
    Loop Until (Selected = True)
    RecognitionMatch(OutputFrog) = SearchFrog
    OutputFrog = OutputFrog + 1
    If OutputFrog >= RecognitionFrog Then
        OutPutOver = True
    End If
Loop Until (OutPutOver = True)
For i = 0 To RecognitionFrog - 1
    tmpwavefile = Trim(Format(Match(RecognitionMatch(i)).speciesid, "00")) & Trim(Str(((i + 1) Mod 3) + 1))
    WAVEfilename(i) = "D:\46 蛙聲辨識\CallData" & tmpwavefile & ".wav"
Next
MyDB.Open "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=D:\46 蛙聲辨識\FrogCall.mdb;Persist Security Info=False" ' 開啟資料庫
MyRec.Open "Frog", MyDB ' 開啟資料庫中的表單 Frog
currentFrog = 0
Do Until MyRec.EOF ' 取出相似度最高的前 4 種蛙類之資料
    Frog(currentFrog).speciesid = MyRec.Fields("speciesid").Value
    Frog(currentFrog).species = MyRec.Fields("species").Value
    Frog(currentFrog).family = MyRec.Fields("family").Value
    Frog(currentFrog).JPGfilename = "D:\46 蛙聲辨識\FrogData" & Trim(MyRec.Fields("filename").Value) & ".jpg"
    Frog(currentFrog).content = MyRec.Fields("content").Value
    currentFrog = currentFrog + 1
    MyRec.MoveNext
Loop
MyRec.Close
MyDB.Close
Text3.Text = "相似度排名 1 的蛙種" ' 輸出辨識結果
Text4.Text = "相似度排名 2 的蛙種"
Text5.Text = "相似度排名 3 的蛙種"
Text6.Text = "相似度排名 4 的蛙種"
Picture3.Picture = LoadPicture(Frog(Match(RecognitionMatch(0)).speciesid - 1).JPGfilename) ' 相似度排名 1 的蛙類圖片
Picture4.Picture = LoadPicture(Frog(Match(RecognitionMatch(1)).speciesid - 1).JPGfilename) ' 相似度排名 2 的蛙類圖片
Picture5.Picture = LoadPicture(Frog(Match(RecognitionMatch(2)).speciesid - 1).JPGfilename) ' 相似度排名 3 的蛙類圖片
Picture6.Picture = LoadPicture(Frog(Match(RecognitionMatch(3)).speciesid - 1).JPGfilename) ' 相似度排名 4 的蛙類圖片
End Sub

```

評 語

040810 識破你的「蛙」言巧語

本作品利用現代人類語音處理技術之能量法與梅爾倒頻譜分析法以發展蛙聲品種辨識系統，在程式測試及樣本收集分析等方面，均著力甚深。本作品之成果足以作為高中教具。團隊之合作精神也頗足鼓勵。

本作品若能在蛙聲辨識特徵之選取，樣本數及採樣頻率等方向多加深入研討，並提出這些相關因素與辨識成效間之關係，則作品將更臻完善。