多媒體技術概論

Final project report

Genre Classification

Group 11

100010018李哲銓

100071049何天與

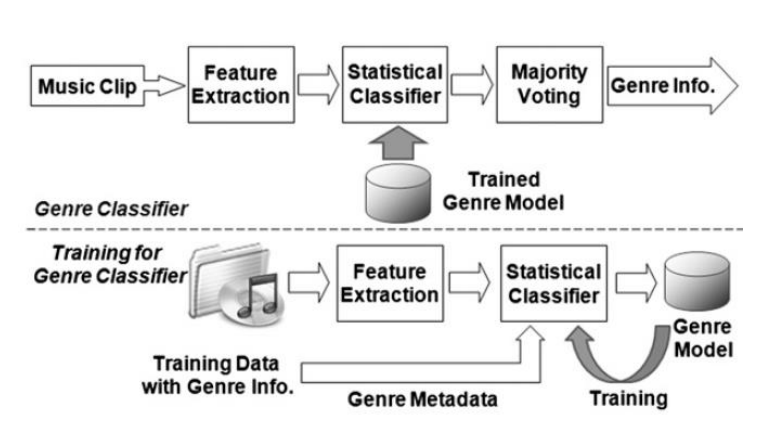
100070023楊雅琪

指導教授

賴尚宏 教授

Genre Classification

我們依照題目投影片上的步驟做training 和 testing



Training:

1. Extract features
2. Classify (use 60 songs to train for each music style)

Testing

1. Majority voting

**Extract features**

我們選擇的features分為三個大項：Timbral Texture Features、Rhythmic Content Features、Pitch Content Features

1. Timbral Texture Features
   1. Zero Crossing Rate：

**定義**：平均每一秒這首歌的訊號在time domian經過0的次數。

**實作**：

參考MIR toolbox中mirzeorcross.m檔，將原本toolbox的output為一object，改為output是一數值，此數值即為Zero Crossing Rate。

* 1. Low Energy Feature：

**定義**：將這首歌在time domain分成許多frame，Low energy Feature值為RMS energy低於平均RMS energy的frame數除以全部的frame數。

**實作**：

Call MIR toolbox中的mirlowenergy(filename ,'Threshold’,t)，t=1，output為一object，經過一些處理取出此首歌low energy feature值。

* 1. Spectral Roll off：

**定義**：在一首歌的頻譜上，Spectral roll off值為一頻率，低於此頻率的頻率的magnitude加總為全部頻率的magnitude加總的85%。

**實作**：

Call MIR toolbox中的mirrolloff(filename) ，output為一object，經過一些處理取出此首歌Roll off值。

* 1. Brightness：

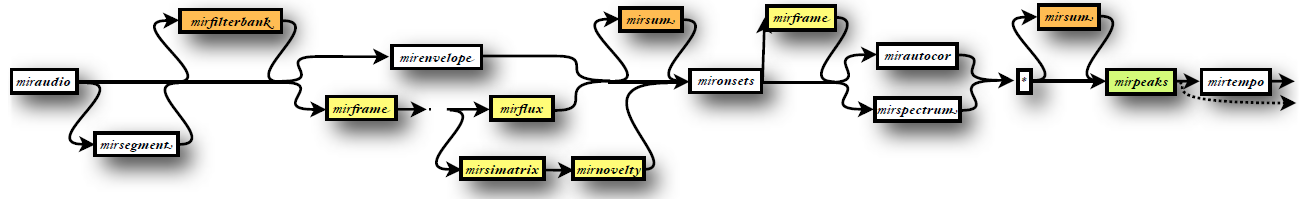
**定義**：在一首歌的頻譜上，給定一頻率(cut off frequency)，計算高於此頻率的所有頻率的energy加總除以頻譜上所有頻率的energy加總。

**實作**：

Call MIR toolbox中的mirbrightness (filename) ，此參數傳入型態cut off frequency為1500Hz，output為一object，經過一些處理取出此首歌Brightness值。

1. Rhythmic Content Features
   1. Tempo

**定義**：以偵測onset detection curve的週期來計算該音頻一分鐘有幾個beat (Beats per minute, BPM)

**實作**：呼叫 Mir toolbox中的 mirtempo() 函式，傳入檔名，output為一object，經過一些處理取出此首歌tempo值。

* 1. Pulse clarity

**定義**：估計rhythm的清晰度，也就是beats的強度

**實作**：呼叫 Mir toolbox中的 mirpulseclarity() 函式，傳入檔名，output為一object，經過一些處理取出此首歌tempo值。

1. Pitch Content Features

我們選擇了Pitch detection裡的七項，並使用frame decomposition，window length=46.4ms、hop factor=10ms (Tolonen and Karjalainen, 2000)，並使用mirpitch(filename,’Frame’,46.4,10);

* 1. Meanfrequency

**定義**：平均一首歌曲的頻率

**實作**：呼叫 Mir toolbox中的 mirpitch() 函式，傳入檔名、frame、window length、hop factor，output為一object，經過一些處理取出此首歌frequency值。

* 1. Slope

**定義**：固定時間內一首歌曲平均波形的斜率

**實作**：呼叫 Mir toolbox中的 mirpitch() 函式，傳入檔名、frame、window length、hop factor，output為一object，經過一些處理取出此首歌Slope值。

* 1. Period Frequency

**定義**：偵測出一首歌的周期後，並擷取此週期的頻率

**實作**：呼叫 Mir toolbox中的 mirpitch() 函式，傳入檔名、frame、window length、hop factor，output為一object，經過一些處理取出此首歌Period frequency值。

* 1. Period Amplitude

**定義**：偵測出一首歌的周期後，並擷取此週期的平均強度

**實作**：呼叫 Mir toolbox中的 mirpitch() 函式，傳入檔名、frame、window length、hop factor，output為一object，經過一些處理取出此首歌Amplitude值。

* 1. Meanfrequency

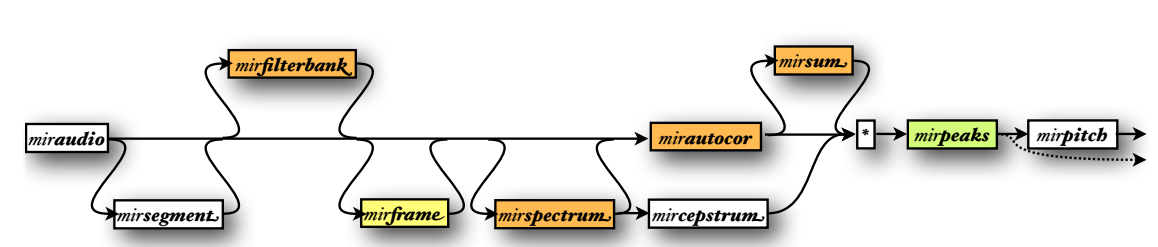
**定義**：平均一首歌曲的頻率

**實作**：呼叫 Mir toolbox中的 mirpitch() 函式，傳入檔名、frame、window length、hop factor，output為一object，經過一些處理取出此首歌frequency值。

* 1. Pitch Period Entropy (PPE)

**定義**：將干擾去除後的頻率

**實作**：呼叫 Mir toolbox中的 mirpitch() 函式，傳入檔名、frame、window length、hop factor，output為一object，經過一些處理取出此首歌較少雜訊的frequency值。

其流程圖：

Method

一開始，我們利用Matlab 的 toolbox - MIR toolbox，並做了一些修正，來提取features。

接下來，我們用dataset (**GTZAN** )中每個音樂類別的前60首歌來做training，產生classification model，剩下的40首歌用來test model。所以最後是用40\*10=400首歌來test model，計算準確度。

對於classify的方法，我們後來決定不再那麼依賴網路上現有的toolbox，決定自己實作，也不是用老師ppt給的兩種方法之一，而是自己想了一個方法：用每個提取feature的function，求出每首用來train的歌的feature值，60首歌可以畫出如以下的折線圖(以function mirtempo()和disco為例)。為了讓分類更準確，對於一個類型下的每一個feature，60首歌會有60個值，我們去掉當中10個最邊緣的值 (如下圖discoTempo所示)。

接著我們取出每個類別剩下的50個值的最大值和最小值，定義該類別的range為(最小值, 最大值)，所以對於每種feature，我們都有10個不同的range對應於10個不同的曲風。Range即為我們用來判斷曲風的標準。

Test時，對於每首歌，一次用兩種曲風來比較。總共7+2+4=13種feature。每首歌test出來的13個值分別跟兩種曲風的13種feature的range比較。例如，如果該首歌的brightness值有落在該曲風的brightness\_range裡面，那該曲風的變數值就加一(score++)。最後比較兩個曲風的變數值哪個較高，則該曲風的vote就加一。比完全部後，計算10個vote的值，該首歌屬於vote最高的那種曲風。



Test Result

目前測試結果尚未出來（目前已跑九小時），因此有結果後，會再補上準確率。