

# Music Information Retrieval HW4 Report

102062209 邱政凱

這次的作業要我們實作的是 Tempo estimation，要從 Ballroom Dataset 之中計算不同種類舞曲的 tempo 和 beats.

## Q1.

第一題的部份是要我們用 Fourier tempogram 的方式把 Dataset 中音樂的 tempo 算出來，並且分別去計算 P Score 和 ALOTC Score。實作的部分我使用的是 matlab 的 tempogram toolbox。步驟上首先就是用 spectral flux 的方式計算出音樂能量的 novelty curve，並且對此 novelty curve 再做一次 STFT 以得到所謂的 Fourier Tempogram。接著，題目要求我們在此 tempogram 上面找出兩個 predominant tempo。我使用的方法是：用 `abs(tempogram)` 取出 tempogram 的絕對強度，接著用 `sum(tempogram,2)` 的方式把所有時間點上對應的 BPM 加總，接著用 `max` 取出這個加總的向量中強度最大的 tempo。第二個 tempo 我的計算方式是把第一個算出來的 tempo，其上下 25% 的 tempogram 都歸零之後在此次用同上的方法取出強度最大的 tempo。這麼做的原因是因為如果不歸零就取對應次大能量的 BPM 的話，基本上只會取到跟最大能量正負一個 index 的 BPM 而已。

```
74 - [MAX Index1] = max(realSum);
75 - t1 = int32(BPM(Index1));
76 - range = double(t1 * 0.25);
77 - downLimit = double(t1 - range);
78 - upLimit = double(t1 + range);
79 - [ ~ ,li] = min(abs(BPM(:) - (downLimit)));
80 - [ ~ ,ui] = min(abs(BPM(:) - (upLimit)));
81 - li = int32(li); ui = int32(ui);
82 - realSum(li:ui) = 0;
83 - % realSum(MAXI) = 0;
84 - [SUB Index2] = max(realSum);
85 - t2 = int32(BPM(Index2));
```

Code 的部分大致上就如上，不過這邊有個重點是，最後從 sum vector 取出來的只是 tempo 的 index，還必須從原本在計算 Fourier Tempogram 時取出來的 row index mapping 回到 BPM 的 Vector 中取出對應的 BPM 才行。

首先用這兩個 tempo 計算 estimation 的 P Score，兩個 tempo 的權重分別對應各自在 tempogram 上的強度，估算的 Tempo 如果介於正確答案的 8% 以內則依照權重計分，而 ALOTC 則只需兩個 tempo 有任一個落在 8% 以內即得一分。

依照每個 genre，在跑每首音樂的時候就把分數累加，算完一整個 genre 之後便除以該 genre 內的音樂數以求得平均分數。(Rumba 的部分我把三種 Rumba

都合到同一個 genre 來計算)

這一題其實一開始讓我有點小挫折，因為算出來的 P Score 非常非常非常的不理想，Cha Cha Cha、Rumba、Samba、Waltz，總共有一半(四種)曲風的 P Score 是 0 分，也就是沒有任何一首的 tempo 估測正確的。以下是輸出的結果：

```
1 ChaChaCha P-Score:0.0072522 36 -----
2 ChaChaCha P-Score(T/2):0.59772 37 Samba P-Score:0
3 ChaChaCha P-Score(T/3):0 38 Samba P-Score(T/2):0.27545
4 ChaChaCha P-Score(T/4):0.38267 39 Samba P-Score(T/3):0.034807
5 ChaChaCha ALOTC-Score:0.018018 40 Samba P-Score(T/4):0.57301
6 ChaChaCha T2/T1 :1.9991 41 Samba ALOTC-Score:0
7 ChaChaCha T1/G :1.9712 42 Samba T2/T1 :1.9123
8 ChaChaCha T2/G :3.9407 43 Samba T1/G :2.1521
9 ----- 44 Samba T2/G :4.0313
10 Jive P-Score:0.44213 45 -----
11 Jive P-Score(T/2):0.31867 46 Tango P-Score:0.15162
12 Jive P-Score(T/3):0.15414 47 Tango P-Score(T/2):0.6356
13 Jive P-Score(T/4):0 48 Tango P-Score(T/3):0.0083732
14 Jive ALOTC-Score:0.83333 49 Tango P-Score(T/4):0.13409
15 Jive T2/T1 :2.0602 50 Tango ALOTC-Score:0.38372
16 Jive T1/G :1.1239 51 Tango T2/T1 :1.9522
17 Jive T2/G :2.2441 52 Tango T1/G :1.4646
18 ----- 53 Tango T2/G :2.8203
19 Quickstep P-Score:0.59707 54 -----
20 Quickstep P-Score(T/2):0.20954 55 VienneseWaltz P-Score:0.55336
21 Quickstep P-Score(T/3):0.008934 56 VienneseWaltz P-Score(T/2):0.41454
22 Quickstep P-Score(T/4):0 57 VienneseWaltz P-Score(T/3):0
23 Quickstep ALOTC-Score:0.96341 58 VienneseWaltz P-Score(T/4):0
24 Quickstep T2/T1 :1.8293 59 VienneseWaltz ALOTC-Score:0.98462
25 Quickstep T1/G :0.97928 60 VienneseWaltz T2/T1 :1.9644
26 Quickstep T2/G :1.7925 61 VienneseWaltz T1/G :1.0033
27 ----- 62 VienneseWaltz T2/G :1.9684
28 Rumba P-Score:0 63 -----
29 Rumba P-Score(T/2):0.54036 64 Waltz P-Score:0.038398
30 Rumba P-Score(T/3):0.0085621 65 Waltz P-Score(T/2):0.46221
31 Rumba P-Score(T/4):0.40862 66 Waltz P-Score(T/3):0.10776
32 Rumba ALOTC-Score:0 67 Waltz P-Score(T/4):0.26737
33 Rumba T2/T1 :2.0001 68 Waltz ALOTC-Score:0.081818
34 Rumba T1/G :1.9672 69 Waltz T2/T1 :1.9621
35 Rumba T2/G :3.9344 70 Waltz T1/G :1.8905
36 ----- 71 Waltz T2/G :3.5022
72 -----
```

Figure(1) First run of Fourier Tempogram algorithm

可以看到八個 genres 裡面有四個 genre 幾乎是完全沒估測對。納悶了很久，赫然發現 matlab 的 tempogram toolbox 中，在運算時是以 22050 Fs 為基準在運算的，而 Ballroom Dataset 原本的 Sampling Frequency 是 44100。我認為也許是因為這樣的關係，造成了 toolbox 中使用的 feature rate(novelty curve 的 sampling rate)因此多了一倍(當然也有可能不是因為這個原因，不過還是基於好奇心做了別的實驗)。於是我把計算 Fourier Tempogram 時使用的 Feature rate 除以 2，發現計算出來的結果比一開始還要好上許多。以下是把 Feature rate 除以 2 之後再照著上面的步驟跑一次之後的結果：

```

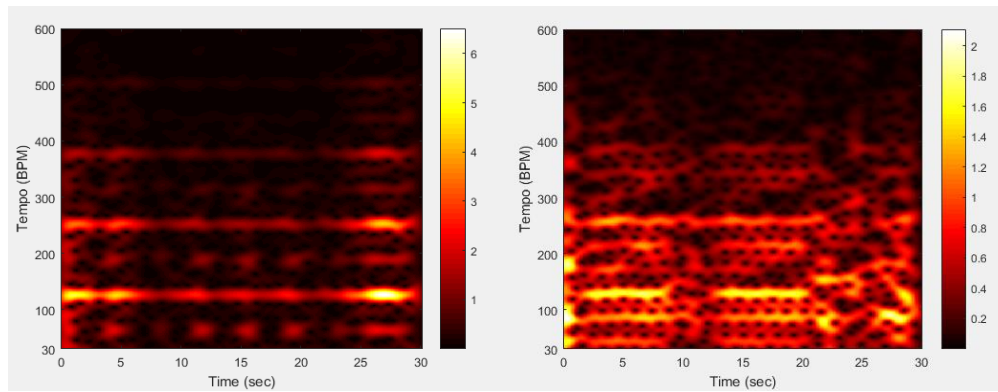
1 ChaChaCha P-Score:0.59603 36 -----
2 ChaChaCha P-Score(T/2):0.38771 37 Samba P-Score:0.27019
3 ChaChaCha P-Score(T/3):0 38 Samba P-Score(T/2):0.56589
4 ChaChaCha P-Score(T/4):0 39 Samba P-Score(T/3):0
5 ChaChaCha ALOTC-Score:0.99099 40 Samba P-Score(T/4):0.014784
6 ChaChaCha T2/T1 :1.9976 41 Samba ALOTC-Score:0.62791
7 ChaChaCha T1/G :0.98619 42 Samba T2/T1 :1.9318
8 ChaChaCha T2/G :1.9701 43 Samba T1/G :1.0842
9 ----- 44 Samba T2/G :2.0605
10 Jive P-Score:0.34175 45 -----
11 Jive P-Score(T/2):0 46 Tango P-Score:0.62602
12 Jive P-Score(T/3):0 47 Tango P-Score(T/2):0.13121
13 Jive P-Score(T/4):0 48 Tango P-Score(T/3):0
14 Jive ALOTC-Score:0.68333 49 Tango P-Score(T/4):0
15 Jive T2/T1 :1.9813 50 Tango ALOTC-Score:1
16 Jive T1/G :0.56248 51 Tango T2/T1 :1.9281
17 Jive T2/G :1.0839 52 Tango T1/G :0.72279
18 ----- 53 Tango T2/G :1.3806
19 Quickstep P-Score:0.22307 54 -----
20 Quickstep P-Score(T/2):0 55 VienneseWaltz P-Score:0.42166
21 Quickstep P-Score(T/3):0 56 VienneseWaltz P-Score(T/2):0
22 Quickstep P-Score(T/4):0 57 VienneseWaltz P-Score(T/3):0
23 Quickstep ALOTC-Score:0.57317 58 VienneseWaltz P-Score(T/4):0
24 Quickstep T2/T1 :1.807 59 VienneseWaltz ALOTC-Score:0.93846
25 Quickstep T1/G :0.48973 60 VienneseWaltz T2/T1 :1.9537
26 Quickstep T2/G :0.88563 61 VienneseWaltz T1/G :0.50222
27 ----- 62 VienneseWaltz T2/G :0.97998
28 Rumba P-Score:0.53614 63 -----
29 Rumba P-Score(T/2):0.40898 64 Waltz P-Score:0.46364
30 Rumba P-Score(T/3):0 65 Waltz P-Score(T/2):0.28093
31 Rumba P-Score(T/4):0 66 Waltz P-Score(T/3):0
32 Rumba ALOTC-Score:0.94898 67 Waltz P-Score(T/4):0
33 Rumba T2/T1 :1.9871 68 Waltz ALOTC-Score:0.86364
34 Rumba T1/G :0.9865 69 Waltz T2/T1 :1.8372
35 Rumba T2/G :1.9613 70 Waltz T1/G :0.99097
36 ----- 71 Waltz T2/G :1.7957
37 ----- 72 -----

```

**Figure(2) Outcome of Fourier tempo estimation using half feature rate**

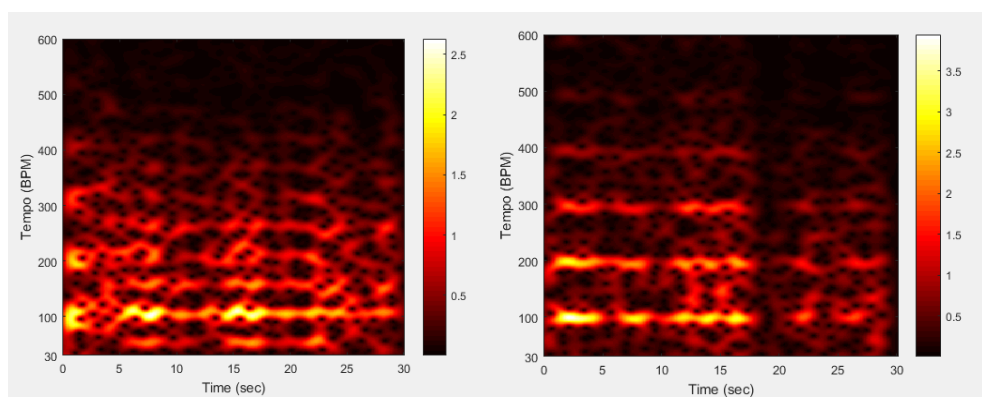
可以很明確的發現，整體的 estimation 表現比前面還有好上許多，除了 Jive / Quickstep / Viennese Waltz 三個 genre 的表現跟前面相比有些許下降外，其他 genre 的表現都來到正常的範圍了。儘管可能這麼做有點改變了 Fourier tempogram estimation 的物理意義，但為求較好的 estimation 結果，接下來幾題我用 Fourier tempogram 計算估測分數時都會使用一半的 feature rate。(比較時我會同時比較修改 feature rate 前後的)。不過不論如何，因為我的算法的關係，我抓取的 T1 跟 T2 勢必會有一段距離，所以用 P Score 來算時基本上一定只會有一個 tempo 是正確的，因此絕對會比 ALOTC 還要來的低許多。

在這邊附上不同種類音樂 CLIP 的範例 tempogram 做個簡單的比較：



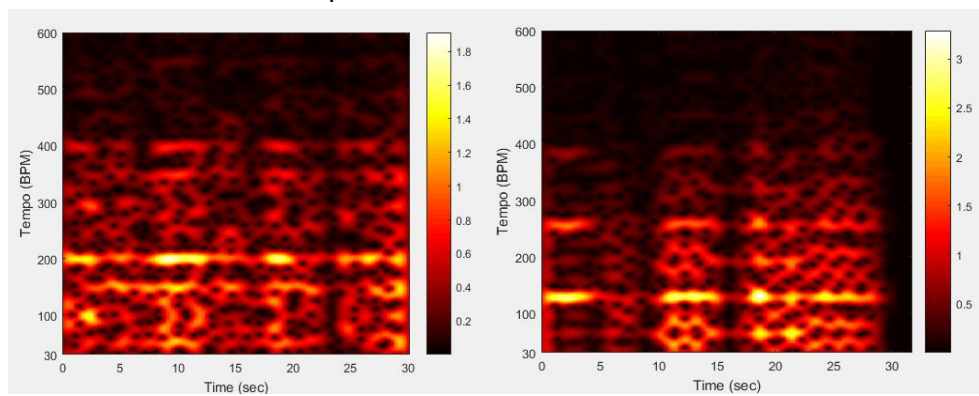
ChaChaCha

Jive



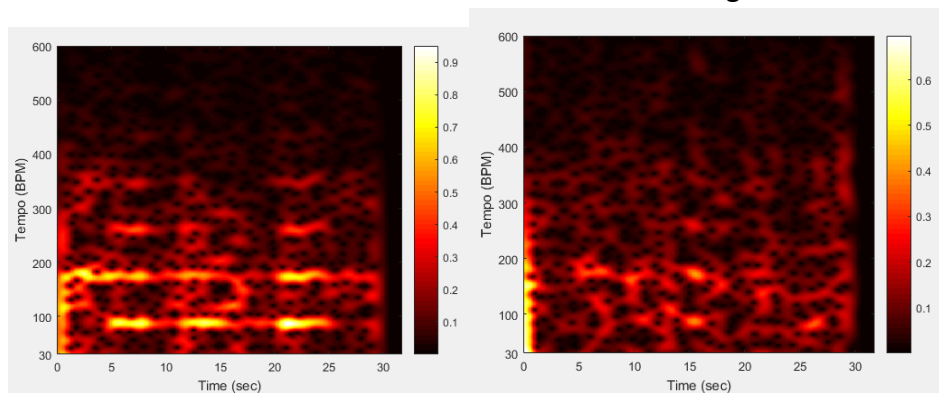
Quickstep

Rumba



Samba

Tango



Viennese Waltz

Waltz

可以發現，ChaChaCha 和 Rumba 的 tempogram 是裡面相對乾淨的(也就是說整個 clip 的 tempo 是穩定的)，造成了較高的 P Score。而 Jive、Quickstep、Samba 的 tempogram 則顯得較為雜亂，可見舞曲的 tempo 可能因為較複雜的配樂而顯得不穩定，而導致較低的 P Score。Viennese Waltz 和 Tango 的 tempogram 雖然稱不上乾淨，但是大致還是整齊的，P Score 介於中間。Waltz 則因為整個 tempogram 強度都偏弱，所以反而正確的 tempo 附近的強度容易凸顯出來，因此 P Score 也不至於太低。

---

## Q2 、

這部分的結果我已經有同時印在 Figure(1)跟 Figure(2)之中了。因為 T2 跟 T1 兩個 predominant tempo 有指定順序(T2 較快 T1 較慢)，我們可以看到，其實不論哪種音樂 genre，T2/T1 幾乎都是等於 2，也就是說最強的兩個 tempo 之間幾乎就是單純的兩倍數的關係。這樣的結果其實完全在意料之中，就跟 Pitch detection 會有 octave equivalence 一樣，用 STFT 計算 Novelty curve 的 tempogram 也很自然會有這種 harmonics / sub harmonics 的問題。不過像是 tempogram 較雜亂的 Quickstep 或 Waltz，比例就會偏離 2 稍微多一點。

T1/G 跟 T2/G 就比較有意思了，從這兩個數值我們可以看到每個 genre 的 tempogram 的不同特性。ChaChaCha、Rumba、Samba、Waltz 是 T1 跟 G 比較接近，而 Jive、Quickstep、Viennese Waltz 則是 T2 跟 G 比較接近 (在沒有改變 feature rate 的狀況下是：前者 T1、T2 分別是 G 的 2、4 倍，而後者的 T1 則吻合正確的 G，T2 是 G 的兩倍)。Tango 則是比較沒有明顯的趨勢。

經觀察可以發現，ChaChaCha、Rumba、Samba、Waltz 四種樂曲的 Tempo 都偏低，而 Jive、Quickstep、Viennese Waltz 則是 Tempo 都偏高。

稍微去聽一下各個樂種的曲子，我覺得解釋方式也許是：容易被判快的樂種，他的背景配樂經常會有速度較快打擊樂、弦樂等伴奏，速度比實際舞曲的 Tempo 還快，因此導致容易被判快。而容易被判慢 (其實在沒有改變 feature rate 的狀況下，這些樂種的 T1 是跟正確 Tempo 吻合的) 的樂種，因為配樂幾乎就是在舞曲的拍點上進行，因此不會被判快。Tango 則比較特別，有時配樂會比較快，有時又會在舞曲的拍點上，因此比較難以歸類到這兩類中。

---

## Q3 、

這部分的結果我也已經有同時印在 Figure(1)跟 Figure(2)之中了。

這部分，用未修改 Feature rate 時的結果來解釋會較為清楚其物理意義。就如同上一題所說，ChaChaCha、Rumba、Samba、Waltz (同時也是在未修改 feature rate 時 P Score 都是 0 的樂種) 容易把 tempo 判快，T1、T2 會變成 G 的

2、4 倍，因此  $T/2$  和  $T/4$  在這些樂種裡面算出來的 P Score 都會特別好，而  $T/2$  還會大於  $T/4$ 。Tango 在這邊也是  $T/2$  的效果最好。

而剩下的 Jive、Quickstep、Viennese Waltz 就如同前面所說， $T1$  大致對應  $G$ ， $T2$  大致對應  $2 \cdot G$ ，因此  $T/2$  在這些樂種上算出來的 P Score 勉強還不會掉到太低，不過  $T/4$  就全部都是 0 了)。 $T/3$  的部分，除了 Jive 和 Waltz 以外其他 P Score 幾乎都是 0，至於 Jive 和 Waltz 相較於其他樂種不一樣的地方也許是他們的 tempogram 比較沒有那麼整齊的長條狀能量分布而是比較鬆散的分布，因此就算不是 sub harmonic，抓出來的 tempo 也許還是能碰巧落在  $G$  的正負 8% 以內吧。

---

Q4、

這題是要用 ACF(Autocorrelation Function)重作第一、二題。

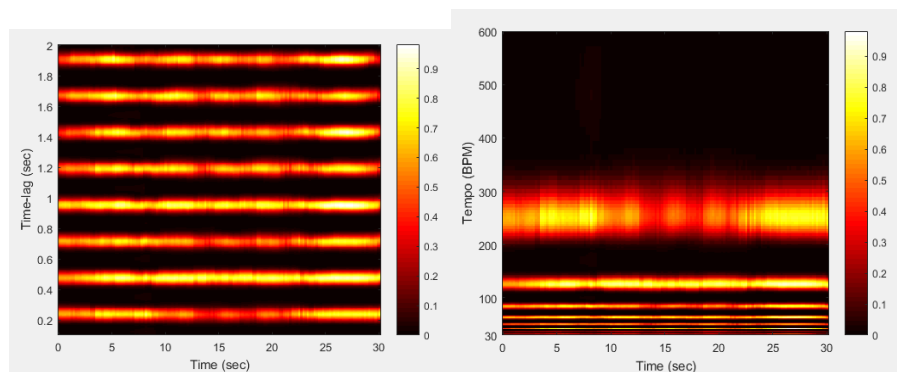
實作的部分，基本上跟第一題大致一樣，不過一開始算出來的 ACF tempogram 是 time-lag 的，所以必須把 y 軸 rescale 轉換成 BPM。另外值得一提的是，用 ACF 實作的話不需修改 Feature rate 效果才會比較 ok，因此 ACF 的部分我就沒有去動 feature rate 了。以下是實驗跑出來的結果：



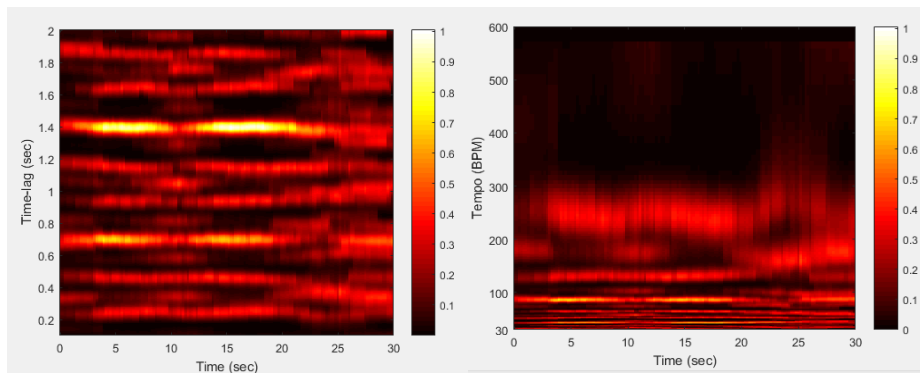
1	ChaChaCha P-Score:0.39366	49	Samba P-Score:0.28133
2	ChaChaCha P-Score(T/2):0.052685	50	Samba P-Score(T/2):0.1949
3	ChaChaCha P-Score(T/3):0	51	Samba P-Score(T/3):0
4	ChaChaCha P-Score(T/4):0	52	Samba P-Score(T/4):0
5	ChaChaCha P-Score(2*T):0.42527	53	Samba P-Score(2*T):0.33423
6	ChaChaCha P-Score(3*T):0	54	Samba P-Score(3*T):0.016168
7	ChaChaCha P-Score(4*T):0.093365	55	Samba P-Score(4*T):0
8	ChaChaCha ALOTC-Score:0.78378	56	Samba ALOTC-Score:0.56977
9	ChaChaCha T2/T1 :2.1594	57	Samba T2/T1 :2.4946
10	ChaChaCha T1/G :0.48439	58	Samba T1/G :0.58201
11	ChaChaCha T2/G :1.0224	59	Samba T2/G :1.3854
12	-----	60	-----
13	Jive P-Score:0.15847	61	Tango P-Score:0.40608
14	Jive P-Score(T/2):0	62	Tango P-Score(T/2):0.032903
15	Jive P-Score(T/3):0	63	Tango P-Score(T/3):0
16	Jive P-Score(T/4):0	64	Tango P-Score(T/4):0
17	Jive P-Score(2*T):0.47038	65	Tango P-Score(2*T):0.31079
18	Jive P-Score(3*T):0	66	Tango P-Score(3*T):0
19	Jive P-Score(4*T):0.37115	67	Tango P-Score(4*T):0.20532
20	Jive ALOTC-Score:0.31667	68	Tango ALOTC-Score:0.81395
21	Jive T2/T1 :2.1338	69	Tango T2/T1 :2.502
22	Jive T1/G :0.31179	70	Tango T1/G :0.43844
23	Jive T2/G :0.65676	71	Tango T2/G :0.99533
24	-----	72	-----
25	Quickstep P-Score:0.20144	73	VienneseWaltz P-Score:0.32088
26	Quickstep P-Score(T/2):0	74	VienneseWaltz P-Score(T/2):0
27	Quickstep P-Score(T/3):0	75	VienneseWaltz P-Score(T/3):0
28	Quickstep P-Score(T/4):0	76	VienneseWaltz P-Score(T/4):0
29	Quickstep P-Score(2*T):0.38988	77	VienneseWaltz P-Score(2*T):0.10147
30	Quickstep P-Score(3*T):0.061593	78	VienneseWaltz P-Score(3*T):0.44377
31	Quickstep P-Score(4*T):0.27109	79	VienneseWaltz P-Score(4*T):0
32	Quickstep ALOTC-Score:0.40244	80	VienneseWaltz ALOTC-Score:0.66154
33	Quickstep T2/T1 :2.3335	81	VienneseWaltz T2/T1 :2.4705
34	Quickstep T1/G :0.32728	82	VienneseWaltz T1/G :0.31709
35	Quickstep T2/G :0.73222	83	VienneseWaltz T2/G :0.78474
36	-----	84	-----
37	Rumba P-Score:0.41495	85	Waltz P-Score:0.40479
38	Rumba P-Score(T/2):0.20597	86	Waltz P-Score(T/2):0.22656
39	Rumba P-Score(T/3):0	87	Waltz P-Score(T/3):0.004858
40	Rumba P-Score(T/4):0	88	Waltz P-Score(T/4):0.0095685
41	Rumba P-Score(2*T):0.18377	89	Waltz P-Score(2*T):0.20289
42	Rumba P-Score(3*T):0.014446	90	Waltz P-Score(3*T):0.028118
43	Rumba P-Score(4*T):0.0098234	91	Waltz P-Score(4*T):0
44	Rumba ALOTC-Score:0.82653	92	Waltz ALOTC-Score:0.8
45	Rumba T2/T1 :2.111	93	Waltz T2/T1 :2.3731
46	Rumba T1/G :0.68043	94	Waltz T1/G :0.72271
47	Rumba T2/G :1.4012	95	Waltz T2/G :1.6023
		96	-----

**Figure(3) Outcome of ACF tempogram**

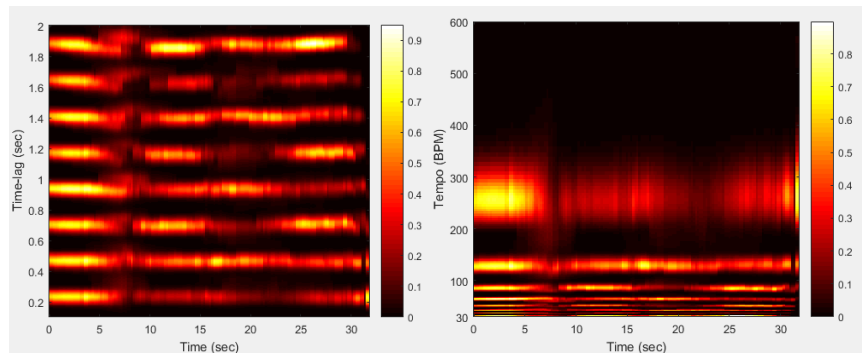
然後以下這邊則選取幾個樂種展示跑出來的 ACF Tempogram，左邊是 time-lag scale，右邊則是 time-BPM。從 Time-lag acf tempogram 中不難看出來，不同於 Fourier Tempogram 的是，他的能量強度在 lag 軸上是循環的，不像 Fourier tempogram 或是轉換成 time-BPM scale 之後的 tempogram。而在 Time-BPM scale 中不像是 Fourier tempogram 會算出很多 harmonics，ACF tempogram 容易算出來的是 sub harmonics。這其實不難解釋，看 ChaChaCha 左側的 time-lag tempogram 就知道，0.25 sec 的 Lag 也可以解釋其實是 0.5 sec、0.75sec 的 Lag，Lag 越久 BPM 自然越低。



**ChaChaCha**



Jive



Tango

在跑出來的 estimation 結果方面，可以發現其實整體結果都比 Figure(2)來的稍差。有趣的是，在 Fourier Tempogram(未修改 Feature rate)中 P Score 較低的樂種，也就是容易被算快的較慢的樂種，在 ACF Tempogram 中反而表現較好，而在 Fourier Tempogram(未修改 Feature rate)中 P Score 較高的樂種，也就是 G 較高的樂種，在 ACF Tempogram 中反而表現較差。(ACF Tempogram 的表現比較近似於我修改過 feature rate 之後的 Fourier tempogram)。

這可以很簡單的解釋為 Fourier tempogram 傾向於找出 harmonics，所以對較慢的樂曲不利，而 ACF 傾向於找出 Sub harmonics，因此對較快的樂曲不利。

T1/T2 的部分，跟 Fourier tempogram 相似，大部分的樂種都是 2 左右，不過有幾個樂種會逼近 2.5。T1/G 跟 T2/G 的部分，可以發現除了 ChaChaCha 跟 Tango 以外幾乎沒有樂種的 T1 或 T2 的平均是穩定落在 G 的附近的，也就是說正確的 tempo 有時是 T1 算出來的，有時是 T2 算出來的，這點跟 Fourier tempogram 時不太一樣，大部分的樂種的正確的 G 的估測幾乎都是由 T1 或 T2 一方獨佔。

Q5、

這部分的結果我也直接印在 Figure(3)了。可以很明顯地看到，不同於 Fourier Tempogram(未修改 Feature rate 的狀況)時，很多樂種較好的 P Score 都是



在  $T/2$  或  $T/4$  時得出來的，ACF Tempogram 傾向於在  $T*2$ (Viennese Waltz 是  $T*3$ ) 時有較好的 P Score 表現。

這部分的解釋其實就跟上面那題一樣，ACF Tempogram 傾向於找出 Sub harmonics，因此傾向於把 tempo 算的太慢，因此把估測的 tempo 乘上倍數可以修正這樣的誤差。(相對於 Fourier tempogram 就是容易算出 harmonics 因此會把 tempo 算的太快需要簡單除  $2/4$  來修正誤差)。而 Viennese Waltz 是用  $3*T$  應該是因為他是  $3/4$  拍的曲風吧！

維也納華爾滋 (Viennese Waltz) 由於風行在奧地利首都維也納，以得名。難能可貴的是，歷經數百年而舞風仍保持原有的風範。因節奏的快速，所以又稱為「快速華爾茲」(Quick Waltz) 即「快三步」舞。對應社交舞裡面的快三步，是國標舞裡面摩登舞的一種，**舞曲配合較快的3/4拍音樂**。維也納華爾滋和華爾滋都是起源於1780年代的維也納，經英國皇家舞蹈協會整理成為標準舞之一。

(取自維基百科)

## Q6、

這題則是要我們想一個可以改進 P Score 的演算法。我試驗了兩種方式，第一種是在每次算 tempogram 時，都同時算出 Fourier tempogram(feature rate 除以二計算出 tempogram 之後修正 column 數)和 Autocorrelation tempogram，然後把兩者做點積。Code 大致上如下：

```
362 -         realF = abs(Ftempogram);
363 -         realF = realF./norm(realF);
364 -         realF(:,2:2:size(realF,2)) = (realF(:,2:2:size(realF,2)) + realF(:,1:2:size(realF,2)-1))/2;
365 -         realF(:,1:2:size(realF,2)) = [];
366 -         realACF = abs(ACFtempogram);
367 -         realACF = realACF ./norm(realACF);
368 -         if size(realF,2) > size(realACF,2),
369 -             sss = (size(realF,2) - size(realACF,2))-1;
370 -             realF(:, (size(realF,2)-sss):size(realF,2)) = [];
371 -         elseif size(realF,2) < size(realACF,2),
372 -             sss = (size(realACF,2) - size(realF,2)) -1;
373 -             realACF(:, (size(realACF,2)-sss):size(realACF,2)) = [];
374 -         end
375 -         realTempogram = realF.*realACF;
```

這樣子結合 Fourier 跟 ACF，我原本的預想是因為一個傾向算出 harmonics，一個傾向算出 sub harmonics，那麼兩者的交集應該就會是正確的 tempo G 了，所以採用點積的方式，不過結果似乎不盡理想，大致如下：

1	ChaChaCha P-Score:0.62403	36	-----
2	ChaChaCha P-Score(T/2):0.30344	37	Samba P-Score:0.29991
3	ChaChaCha P-Score(T/3):0	38	Samba P-Score(T/2):0.5477
4	ChaChaCha P-Score(T/4):0	39	Samba P-Score(T/3):0
5	ChaChaCha ALOT-Score:0.99099	40	Samba P-Score(T/4):0.013978
6	ChaChaCha T2/T1 :1.997	41	Samba ALOT-Score:0.68605
7	ChaChaCha T1/Q :0.90972	42	Samba T2/T1 :2.1526
8	ChaChaCha T2/Q :1.8165	43	Samba T1/Q :0.9914
9	-----	44	Samba T2/Q :2.0263
10	Jive P-Score:0.34661	45	-----
11	Jive P-Score(T/2):0	46	Tango P-Score:0.62792
12	Jive P-Score(T/3):0	47	Tango P-Score(T/2):0.077044
13	Jive P-Score(T/4):0	48	Tango P-Score(T/3):0
14	Jive ALOT-Score:0.73333	49	Tango P-Score(T/4):0
15	Jive T2/T1 :2.1004	50	Tango ALOT-Score:1
16	Jive T1/Q :0.42824	51	Tango T2/T1 :2.2786
17	Jive T2/Q :0.8813	52	Tango T1/Q :0.58689
18	-----	53	Tango T2/Q :1.237
19	Quickstep P-Score:0.24073	54	-----
20	Quickstep P-Score(T/2):0	55	VienneseWaltz P-Score:0.40165
21	Quickstep P-Score(T/3):0	56	VienneseWaltz P-Score(T/2):0
22	Quickstep P-Score(T/4):0	57	VienneseWaltz P-Score(T/3):0
23	Quickstep ALOT-Score:0.64634	58	VienneseWaltz P-Score(T/4):0
24	Quickstep T2/T1 :1.957	59	VienneseWaltz ALOT-Score:0.87692
25	Quickstep T1/Q :0.44371	60	VienneseWaltz T2/T1 :1.9785
26	Quickstep T2/Q :0.86719	61	VienneseWaltz T1/Q :0.47795
27	-----	62	VienneseWaltz T2/Q :0.94597
28	Rumba P-Score:0.55451	63	-----
29	Rumba P-Score(T/2):0.37547	64	Waltz P-Score:0.4682
30	Rumba P-Score(T/3):0	65	Waltz P-Score(T/2):0.27009
31	Rumba P-Score(T/4):0	66	Waltz P-Score(T/3):0
32	Rumba ALOT-Score:0.94898	67	Waltz P-Score(T/4):0
33	Rumba T2/T1 :1.99	68	Waltz ALOT-Score:0.84545
34	Rumba T1/Q :0.96086	69	Waltz T2/T1 :1.9626
35	Rumba T2/Q :1.9152	70	Waltz T1/Q :0.84103
36	-----	71	Waltz T2/Q :1.6306
		72	-----
		73	

可以看到整體的 P Score 是有“微幅”的上升，不過基本上是不能說有明顯的改善的。原因的部分我在想會不會是跟 Normalization 有關，不過似乎把兩個 tempogram 都 normalize 並不會有改善。

另一個方法則有點算是利用了對於樂種的先備知識。首先我們觀察 Fiugre(2)，也就是把 Feature rate 除以 2 之後計算 Fourier tempogram 的 outcome。我們這部分利用的就是它大部分的樂種幾乎都是 T1 和 T2 某個 estimation 佔了全部的正確估測值，因此，觀察不同樂種 T1/G 和 T2/G 哪個是相當接近於 1.0 的，我們在 estimate P Score 時使用的權重 S1 和 S2，就直接把對應該樂種的權重設成 1，另一個設成 0(例如 ChaChaCha 因為 T1 比較接近，那 ChaChaCha 的 S1 就是 1，S2 就是 0；Jive 則是反過來)。結果大致如下：

1	ChaChaCha P-Score:0.81982	36	-----
2	ChaChaCha P-Score(T/2):0	37	Samba P-Score:0.65116
3	ChaChaCha P-Score(T/3):0	38	Samba P-Score(T/2):0.046512
4	ChaChaCha P-Score(T/4):0	39	Samba P-Score(T/3):0
5	ChaChaCha ALOTc-Score:0.99099	40	Samba P-Score(T/4):0
6	ChaChaCha T2/T1 :1.997	41	Samba ALOTc-Score:0.68605
7	ChaChaCha T1/Q :0.90972	42	Samba T2/T1 :2.1526
8	ChaChaCha T2/Q :1.8165	43	Samba T1/Q :0.9914
9	-----	44	Samba T2/Q :2.0263
10	Jive P-Score:0.73333	45	-----
11	Jive P-Score(T/2):0	46	Tango P-Score:0.5
12	Jive P-Score(T/3):0	47	Tango P-Score(T/2):0.11628
13	Jive P-Score(T/4):0	48	Tango P-Score(T/3):0
14	Jive ALOTc-Score:0.73333	49	Tango P-Score(T/4):0
15	Jive T2/T1 :2.1004	50	Tango ALOTc-Score:1
16	Jive T1/Q :0.42824	51	Tango T2/T1 :2.2786
17	Jive T2/Q :0.8813	52	Tango T1/Q :0.58689
18	-----	53	Tango T2/Q :1.237
19	Quickstep P-Score:0.64634	54	-----
20	Quickstep P-Score(T/2):0	55	VienneseWaltz P-Score:0.87692
21	Quickstep P-Score(T/3):0	56	VienneseWaltz P-Score(T/2):0
22	Quickstep P-Score(T/4):0	57	VienneseWaltz P-Score(T/3):0
23	Quickstep ALOTc-Score:0.64634	58	VienneseWaltz P-Score(T/4):0
24	Quickstep T2/T1 :1.957	59	VienneseWaltz ALOTc-Score:0.87692
25	Quickstep T1/Q :0.44371	60	VienneseWaltz T2/T1 :1.9785
26	Quickstep T2/Q :0.86719	61	VienneseWaltz T1/Q :0.47795
27	-----	62	VienneseWaltz T2/Q :0.94597
28	Rumba P-Score:0.89796	63	-----
29	Rumba P-Score(T/2):0	64	Waltz P-Score:0.57273
30	Rumba P-Score(T/3):0	65	Waltz P-Score(T/2):0
31	Rumba P-Score(T/4):0	66	Waltz P-Score(T/3):0
32	Rumba ALOTc-Score:0.94898	67	Waltz P-Score(T/4):0
33	Rumba T2/T1 :1.99	68	Waltz ALOTc-Score:0.84545
34	Rumba T1/Q :0.96086	69	Waltz T2/T1 :1.9626
35	Rumba T2/Q :1.9152	70	Waltz T1/Q :0.84103
36	-----	71	Waltz T2/Q :1.6306
37	Samba P-Score:0.65116	72	-----
38	Samba P-Score(T/2):0.046512	73	
39	Samba P-Score(T/3):0		
40	Samba P-Score(T/4):0		
41	Samba ALOTc-Score:0.68605		
42	Samba T2/T1 :2.1526		
43	Samba T1/Q :0.9914		
44	Samba T2/Q :2.0263		
45	-----		

可以看到，整體的 P Score 表現都有了驚人的提升。不過這種方法如果沒有先對樂種經過預處理得知其特性的話是不能使用的，所以雖然效果很好不過實際可以應用的範圍應該有限。

## Bonus、

Bonus 的部分做的是 Beat tracking，並且計算不同樂種 tracking 的 F Score。跟 tempo estimation 一樣，這邊我使用的也是 tempogram toolbox，並用 PLP 算法來進行。不同於 tempo estimation 算的是 novelty curve 的 frequency，PLP 算的是 novelty curve 的 Phase。

PLP 的計算方式是先算出 tempogram，接著把 tempogram 的值轉換成 phase，並且計算某時間點對應 tempogram 上最大強度的 tempo 在該時間點上應該會出現的 phase，並且跟前面直接從 tempogram 上算出來的 phase 做比較，若吻合則 PLP curve 的值就會較高。算出 PLP curve 之後，找出所有 curve 上

的 local maximum ,

```
pos = [PLP,PLP(end)] > [PLP(1),PLP];  
neg = ~pos;  
peaks = find(pos(1:end-1).*neg(2:end));
```

把這些點根據 feature rate 和 sampling frequency 轉換成對應的時間，並且跟 annotation 上的時間點做比對，若落於正負 0.07 秒以內則算做正確的 beat。

我的作法是先把 annotation 的值都抓進 list 裡，接著把 PLP 算出來的 beat 點也放進一個 list 裡，接著遍歷 PLP 算出來的 beat list，一旦 annotation 裡有對應的(+0.07 sec)beat，就把它從 annotation 中消除，並且 TP += 1，如果沒有，則 FP +=1，遍歷完之後 annotation 中若還有剩下來的 beat，則算做 FN。Code 如下。

```
while ~feof(beatID),  
    beatTime = fscanf(beatID,'%f',1);  
    [~,] = fscanf(beatID,'%d',1);  
    beats = [beats ; beatTime];  
end  
TP = 0; FP = 0; FN = 0;  
%disp(num2str((length(estimateBeatTime) - length(beats))));  
for idx = 1 : length(estimateBeatTime),  
    if length(beats)>=1,  
        [minValue , minIndex ] = min(abs(beats(:) - estimateBeatTime(idx)));  
        if minValue<0.07,  
            TP = TP + 1;  
            beats(minIndex) = [];  
        else  
            FP = FP + 1;  
        end  
    else  
        FP = FP + 1;  
    end  
end  
end
```

而做出來的結果大致如下：

```
1 ChaChaCha Precision:0.46696  
2 ChaChaCha Recall:0.99737  
3 ChaChaCha F-Score:0.63558  
4 Jive Precision:0.61601  
5 Jive Recall:0.93337  
6 Jive F-Score:0.7265
```

```
1 Quickstep Precision:0.84663  
2 Quickstep Recall:0.94276  
3 Quickstep F-Score:0.88979  
4 Rumba Precision:0.42845  
5 Rumba Recall:0.98242  
6 Rumba F-Score:0.59304  
7 Samba Precision:0.25995  
8 Samba Recall:0.97469  
9 Samba F-Score:0.40767
```

```
1 Tango Precision:0.47797
2 Tango Recall:0.97829
3 Tango F-Score:0.64166
4 VienneseWaltz Precision:0.77854
5 VienneseWaltz Recall:0.95627
6 VienneseWaltz F-Score:0.85381
7 Waltz Precision:0.34887
8 Waltz Recall:0.90078
9 Waltz F-Score:0.49955
10 |
```

其實可以看出來，F Score 較高的樂種也是前面提到所謂的 **tempo** 較快，而配樂基本上都在拍點上的曲子，而配樂不會剛好在拍點上的其他曲風則會導致較低的 F Score。