

2022 年臺灣國際科學展覽會 優勝作品專輯

作品編號	190005
參展科別	電腦科學與資訊工程
作品名稱	一種新的複音音樂片段相似性度量
得獎獎項	四等獎

就讀學校 臺北市立建國高級中學

指導教師 蘇黎、王鼎中

作者姓名 黃家棟

關鍵詞 複音音樂、點對應、相似性

作者簡介



我是黃家棟，就讀臺北市立建國高中。音樂與資訊是我的興趣，能夠寫程式分析音樂，讓我感到驚奇不已，為了解釋數據常常苦思許久，往往發現更多未注意的音樂細節。希望未來也能繼續努力探索、前進。

摘要

平常聽音樂時經常有種似曾相識的感覺。為了描述這種感覺，我們展開了複音音樂片段相似性度量的研究。因為曾經使用過最長公共子序列實作卻效果不如預期，我們將音樂片段正規化後，視為座標平面上的時間、音高點對的集合，使用點對應與二分圖匹配的方法，定義兩個複音音樂片段的相似度為最大權重匹配的平均邊權。我們計算了資料集(JKUPDD)中相同、相異的音樂片段的相似性，調整算法中的參數，找出最適合的參數組合，並且透過音符之間的權重，畫出自相似度矩陣，發現樂曲中的重複片段。

Abstract

It is common for people hearing certain kinds of melody to have a sense of familiarity. To describe the feeling, we began to study on polyphonic music similarity. We used to employ Longest Common Subsequence to realize a music similarity measure but it performed not well enough. We thus designed a new approach, normalizing the onset and pitch of music segments and using point matching and bipartite graph matching to update the weights between notes. We simply define the similarity between two patterns is the average weight of their maximum weight matching. We also calculated the similarity between patterns on JKUPDD to find the best combination of parameters in our algorithms. We then plotted the self-similarity matrix with the weights between notes, which allowed us to discover the repeated patterns in a music score.

壹、前言

一、研究動機

聽音樂時，經常有種似曾相識的熟悉感，覺得聽到的音樂與記憶中的某段旋律很像，想要建立一種相似性度量來描述音樂的相似性。之前研究透過最長公共子序列的方式效果不佳，推測可能是一維的資料表示法造成的音樂資訊損失，因此想要使用二維的資料表示法嘗試改進複音音樂的相似性度量。

二、研究目的

- (一) 建立基於點對應的複音音樂片段相似性度量
- (二) 發現複音音樂中的重複片段

三、研究問題

Discovery of Repeated Themes & Sections 是 MIREX 2017 的任務，演算法需要輸入一段音樂，找出其中的片段與事件。(事件是與片段非常相似的重複)。

貳、研究方法或過程

一、研究設備與器材

(一) 硬體設備

- 1. 個人筆電 CPU Intel Core i5

(二) 軟體環境

- 1. Python: Spyder
- 2. Microsoft Office: Excel

二、研究構思與架構



圖 1. 研究流程圖

三、研究過程

本研究想要將音樂片段視為座標平面上的時間、音高點對，建立一種基於點對應的複音音樂片段相似性度量。這種相似性度量要滿足對音樂片段在音高上平移的不變性，以及在時間方面伸縮的不變性。也就是說，不同調性的音樂片段要被認為是相同的，不同速度演奏的音樂片段也要被認為是相同的。因此我們在資料預處理時，將音樂片段中的每一個音符的開始時間與音高作正規化，使得資料表示法可以滿足以上的不變性。

我們採用了符合幾何值觀的方式定義我們的音樂相似性度量，也就是當兩個音樂片段經過正規化，畫成散佈圖時，兩堆點「重疊度越高」，兩個音樂片段的相似度就越高。為了描述兩音樂片段的「重疊度」，我們使用了點對應的方法，修改 Xiao Liu, Congying Han, 與 Tiande Guo (2015)[6]所提出的公式，賦予兩音符一個權重代表兩者的對應程度。

為了驗證我們提出的相似性度量的有效性，我們使用了 JKUPDD(資料集)，其中包含許多作曲家的複音音樂片段(具代表性的音樂片段叫做「片段」(pattern)，這種片段在樂曲中重複出現許多次，每一次稱為一個在該時間點的事件(occurrence)，我們將計算各個音樂家的音樂片段之間的相似性，分為相同音樂家中，相同片段(pattern)中的第一個與第二個事件(occurrence)，與不同片段(pattern)中的第一個事件(occurrence)。

相同片段(pattern)中的第一個與第二個事件(occurrence)：JKUPDD 中的第一個事件(occurrence)是按照開始時間(onset)排序的，因此，相同片段(pattern)中的第一個事件(occurrence)與最後一個事件(occurrence)不一定有高的相似度，可能因為音符數量差異導致相似度過低。在這個任務中，我們希望前兩個事件(occurrence)的相似度可以盡量高，這樣表示我們的相似性度量可以將「相似」的片段認為是「相似的」。

不同片段(pattern)中的第一個事件(occurrence)：相同音樂家有許多個片段(pattern)，這些片段(pattern)被區分為不同代表著它們之間有根本上的差異，在這個任務中，我們希望相異片段(pattern)的第一個事件(occurrence)計算出的相似度較低，這樣表示我們可以區分出音樂中的「相異」片段(pattern)。

我們設計了一套基於點對應算法的權重更新的方法，應用於 MIREX 2017 的 Discovery of Repeated Themes & Sections。這是一個需要從給定曲子中找出標籤片段(ground truth pattern)與對應的事件(occurrence)的任務，使用 JKUPDD 中五位音樂家的曲子以及對應的標籤片段(ground truth pattern)。

評估的方式為計算三個度量：F1 值(F1 score)、精確率(Precision)、召回率(Recall)。其中評估的面向有四個：standard、establishment、occurrence 以及 three layer。standard 是希望演算法找到與標籤片段一模一樣的片段與事件，occurrence 則是希望演算法可以找到所有的事件。關於此四個面向的度量的定義如下[5]：

假設資料集有 n_p 個標籤片段 $\Pi = \{P_1, P_2, \dots, P_{n_p}\}$ ，演算法找到 n_q 個片段 $\Xi = \{Q_1, Q_2, \dots, Q_{n_q}\}$ ，如果演算法找到資料集中 k 個片段，standard precision(簡記為 P_{std})、standard recall(簡記為 R_{std})、standard F1score(簡記為 $F1_{std}$)定義為 $P_{std} = \frac{k}{n_q}$ 、 $R_{std} = \frac{k}{n_p}$ 、 $F1_{std} = \frac{2P_{std}R_{std}}{P_{std}+R_{std}}$ 。

在定義 establishment 的度量前，要先定義 cardinality 與 establishment matrix。假設資料集有 n_p 個標籤片段 $\Pi = \{P_1, P_2, \dots, P_{n_p}\}$ ，演算法找到 n_q 個片段 $\Xi = \{Q_1, Q_2, \dots, Q_{n_q}\}$ cardinality，輸入為兩個事件。

$$s(P_i, Q_j) = \frac{|P_i \cap Q_j|}{\max\{|P_i|, |Q_j|\}}$$

若 cardinality 的輸入為兩個片段，則代表以下矩陣

$$s(P, Q) = \begin{pmatrix} s(P_1, Q_1) & \cdots & s(P_1, Q_{n_q}) \\ \vdots & \ddots & \vdots \\ s(P_{n_p}, Q_1) & \cdots & s(P_{n_p}, Q_{n_q}) \end{pmatrix}$$

我們將 $s(P, Q)$ 的最大值記為 $S(P, Q)$ ，即 $S(P, Q) = \max\{s(P, Q)\}$ 。

我們用 $S(\Pi, \Xi)$ 表示以下矩陣(稱為 establishment matrix)：

$$S(\Pi, \Xi) = \begin{pmatrix} S(P_1, Q_1) & \cdots & S(P_1, Q_{n_q}) \\ \vdots & \ddots & \vdots \\ S(P_{n_p}, Q_1) & \cdots & S(P_{n_p}, Q_{n_q}) \end{pmatrix}$$

establishment precision、establishment recall 與 establishment F1 score 的定義如下：

$$P_{\text{est}} = \frac{1}{n_q} \sum_{j=1}^{n_q} \max\{S(P_i, Q_j) | i = 1, \dots, n_p\}$$

$$R_{\text{est}} = \frac{1}{n_p} \sum_{i=1}^{n_p} \max\{S(P_i, Q_j) | j = 1, \dots, n_q\}$$

$$F1_{\text{est}} = \frac{2P_{\text{est}}R_{\text{est}}}{P_{\text{est}} + R_{\text{est}}}$$

在定義 occurrence 的度量前，要先定義 occurrence matrix $O(\Pi, \Xi)$ ：

$$O(\Pi, \Xi)_{ij} = \begin{cases} p(s(P_i, Q_j)) & \text{if } S(\Pi, \Xi)_{ij} > 0.75 \\ 0 & \text{else} \end{cases}$$

occurrence precision、occurrence recall 與 occurrence F1 score 則分別定義為 $O(\Pi, \Xi)$ 的 precision、recall 與 F1 score，即

$$P_{\text{occ}} = \frac{1}{n_q} \sum_{j=1}^{n_q} \max\{O_{ij} | i = 1, \dots, n_p\}$$

$$R_{\text{occ}} = \frac{1}{n_p} \sum_{i=1}^{n_p} \max\{O_{ij} | j = 1, \dots, n_q\}$$

$$F1_{\text{occ}} = \frac{2P_{\text{occ}}R_{\text{occ}}}{P_{\text{occ}} + R_{\text{occ}}}$$

在介紹 three layer 的度量之前，我們需要先介紹對於事件的 precision、recall 與 F1 score，當輸入為事件時：

$$F_1(P, Q) = \frac{2P_1(P, Q)R_1(P, Q)}{P_1(P, Q) + R_1(P, Q)}$$

其中

$$P_1(P, Q) = \frac{|P \cap Q|}{|Q|}$$

$$R_1(P, Q) = \frac{|P \cap Q|}{|P|}$$

當輸入為片段時，

$$F_1(P, Q) = \begin{pmatrix} F_1(P_1, Q_1) & \cdots & F_1(P_1, Q_{m_Q}) \\ \vdots & \ddots & \vdots \\ F_1(P_{m_P}, Q_1) & \cdots & F_1(P_{m_P}, Q_{m_Q}) \end{pmatrix}$$

$$P_2 = \frac{1}{m_Q} \sum_{j=1}^{m_Q} \max\{F_1(P_i, Q_j) | i = 1, \dots, m_P\}$$

$$R_2 = \frac{1}{m_P} \sum_{i=1}^{m_P} \max\{F_1(P_i, Q_j) | j = 1, \dots, m_Q\}$$

$$F_2(P, Q) = \frac{2P_2(P, Q)R_2(P, Q)}{P_2(P, Q) + R_2(P, Q)}$$

我們將用 $F_2(\Pi, \Xi)$ 表示以下矩陣：

$$F_2(\Pi, \Xi) = \begin{pmatrix} F_2(P_1, Q_1) & \cdots & F_2(P_1, Q_{n_Q}) \\ \vdots & \ddots & \vdots \\ F_2(P_{n_P}, Q_1) & \cdots & F_2(P_{n_P}, Q_{n_Q}) \end{pmatrix}$$

並定義其 precision、recall、F score 為 three layer 的 precision、recall、F score，即

$$P_3(\Pi, \Xi) = \frac{1}{n_Q} \sum_{j=1}^{n_Q} \max\{F_2(P_i, Q_j) | i = 1, \dots, n_P\}$$

$$R_3(\Pi, \Xi) = \frac{1}{n_P} \sum_{i=1}^{n_P} \max\{F_2(P_i, Q_j) | j = 1, \dots, n_Q\}$$

$$F_3(P, Q) = \frac{2P_3(P, Q)R_3(P, Q)}{P_3(P, Q) + R_3(P, Q)}$$

實作的流程可以分為五個步驟，分別為：資料預處理、計算權重、閾值處理、音符分群、片段分群。

資料預處理：將音樂載入後，對開始時間與音高進行正規化。計算權重：將整首樂曲視為一個音樂片段 P，計算 P 與 P 之間的二分圖權重。閾值處理：針對上三角矩陣中每一個列將權重大於 0.8 的音符取出，作為該音符的音符相似群。音符分群：我們改變[4]中的音符分群、片段分群參數，將 cardinality 大於 0.7 的音符相似群合併，並且透過 min_notes(一個事件的最少音符數)與 adj_thr(兩個事件之間的時間差距)來將音符相似群內分成若干個片段。我們使用片段分群：將高 compactness(緊密程度)的事件篩選出來，並且進行 DBSCAN，即得到各個片段(pattern)與對應的事件(occurrence)。

涉及的相關名詞與概念如下：

(一)相關概念

1. 完全二分圖(complete bipartite graph)

二分圖的頂點由兩個互斥的點集構成，使得所有的邊都是連結分屬兩點集中的兩點，而同一點集內則沒有連邊。當二分圖中對兩點集中的任兩點皆有連邊時，稱為完全二分圖。

2. 完美匹配(perfect matching)

一張圖的匹配是其邊的集合，使得裡面的邊沒有公共的頂點。一個包含圖中所有頂點的匹配，稱為該圖的完美匹配。

3. 最大權重匹配

最大權重匹配是指權重總和最大的匹配。

4. 點對應(point matching)

點對應是在兩個點集中找到彼此對應的點對。Xiao Liu, Congying Han, 與 Tiande Guo (2015) [6] 建立點集之間每個點的權重，利用賦權圖(weighted graph)的最大權重匹配實作點對應。

5. 音樂相似性度量(music similarity measure)

音樂相似性度量是一種用來衡量音樂相似性的公式，使得相似度值介在[0,1]。

6. DBSCAN(Density-based spatial clustering of applications with noise)

DBSCAN 是基於密度的聚類演算法，我們使用 sklearn 的套件進行實作，半徑為 0.3，最少的成員數為 2。

7. 緊密程度(compactness)

緊密程度[1, 2]被認為與音樂片段有關，並且至少會有一個片段的事件具有高的緊密程度，一個事件的緊密程度定義為其音符的數量與所張出的區域中音符數量的比值。

8. 緊密程度閾值(compactness threshold)

緊密程度閾值為x代表被演算法找到的事件，其緊密程度需要比全體事件的緊密程度平均高x個標準差，才會進行片段分群。(簡記為 cmp_thr)。

9. 自相似矩陣(self-similarity matrix)

我們將音樂家的整首樂曲計算每個音之間的權重形成的矩陣稱為自相似矩陣。

(二)資料集與資料表示法

1. JKUPDD(資料集)

包含五位音樂家的音樂片段：Bach、Beethoven、Chopin、Gibbons、Mozart，全名為 Johannes Kepler University Patterns Development Database，是 MIREX 在 Discovery of Repeated Themes & Sections 提供的資料集。其資料格式包含每個音的開始時間(onset)、MIDI number、morphetic pitch、音符時長(duration)、staff number，其中時間皆以四分音符(crotchet note)為單位。

2. 資料表示法

本研究使用正規化的開始時間-音高對(onset-pitch pairs)作為音樂片段的資料表示。對於一個音樂片段P中的一個音p， $\mathbf{p} = (p_{onset}, p_{pitch})$ ，其中 $p_{onset} = \frac{p_{onset} - \min_{\mathbf{p}}\{onsets\}}{\max_{\mathbf{p}}\{onsets\} - \min_{\mathbf{p}}\{onsets\}}$ 、 $p_{pitch} = \frac{p_{pitch} - \min_{\mathbf{p}}\{pitches\}}{\max_{\mathbf{p}}\{pitches\} - \min_{\mathbf{p}}\{pitches\}}$ 。

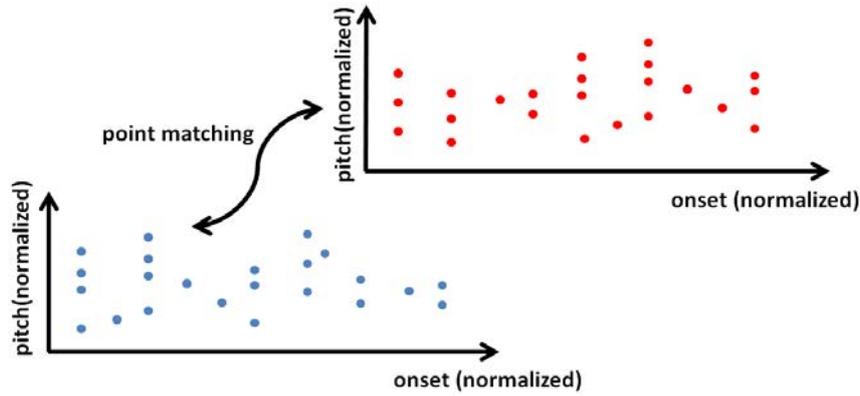


圖 2. 將兩個音樂片段視為不同點集實作點對應

(三) 演算法與相似性度量

1. 權重更新

本研究使用的權重使用迭代的算法公式如下：

$$W_{ij}^{t+1} = W_{ij}^t + \sum_{k,l} W_{kl}^t \cdot s_T(T_{ij}, T_{kl})$$

(1) 權重

權重 W_{ij} 代表音符 $p_i \in P$ 與音符 $q_j \in Q$ 的對應分數(matching score)，權重越大代表 p_i 與 q_j 越適合對應在一起。 W_{ij}^t 表示經過 t 次迭代後的對應分數，其中對所有的 $p_i \in P, q_j \in Q$ ， $W_{ij}^0 = 1$ 。

並且在每次迭代完成以後會進行正規化 $W_{ij}^t = \frac{W_{ij}^t - \min_W\{W\}}{\max_W\{W\} - \min_W\{W\}}$ ，使得 $W_{ij}^t \in [0,1]$ 。

(2) 變換與變換相似度

變換 T_{ij} 代表 p_i 與 q_j 形成的向量，即 $T_{ij} = q_j - p_i$ 。變換 T_{ij} 與 T_{kl} 的相似度 $s_T(T_{ij}, T_{kl})$ 用來衡量 T_{ij} 與 T_{kl} 的相似程度，公式為

$$s_T(T_{ij}, T_{kl}) = 1 - \frac{|T_{ij} - T_{kl}|}{2\sqrt{2}}$$

(3) 鄰居

對於一個音符 $p_i \in P$ (或是 $q_j \in Q$)，其鄰居為在 P (或 Q) 中，距離 p_i (或 q_j) 最近的 K 的音符，

其中 K 為鄰居數量，距離是歐式距離 $\text{dist}(p_i, p_k) = \sqrt{(p_{i_{onset}} - p_{k_{onset}})^2 + (p_{i_{pitch}} - p_{k_{pitch}})^2}$ ，迭代公式中的 p_k 為 p_i 的鄰居， q_l 為 q_j 的鄰居。在更新權重的時候，是以對應的鄰居進行計算。

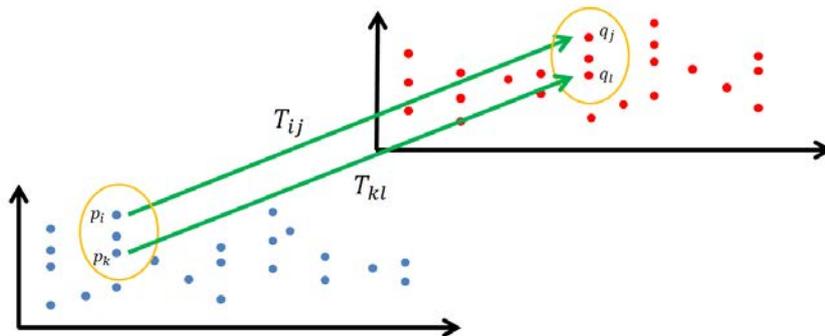


圖 3. 變換 T_{ij} 與鄰居示意圖

2. 二分圖匹配

本研究使用 `scipy` 的 `optimize.linear_sum_assignment` 函式進行二分圖的最大權重匹配。

3. 相似性度量

本研究將兩個音樂片段 P, Q 視為相異的點集，構成完全二分圖 $G(P + Q, W)$ ，其中 W 為 G 的有權重的邊集。並定義兩音樂片段的相似度 $s(P, Q)$ 為其最大權重匹配的平均邊權，並將邊權除以除以 $\max\{n, m\}$ 以應對兩音樂片段的音符數量差距太大造成過高的相似度值。

$$s(P, Q) = \frac{\max\{\sum_{(i,j) \in M} W_{ij}\}}{\max\{n, m\}}$$

其中 n, m 分別為音樂片段 P, Q 的音符數量。

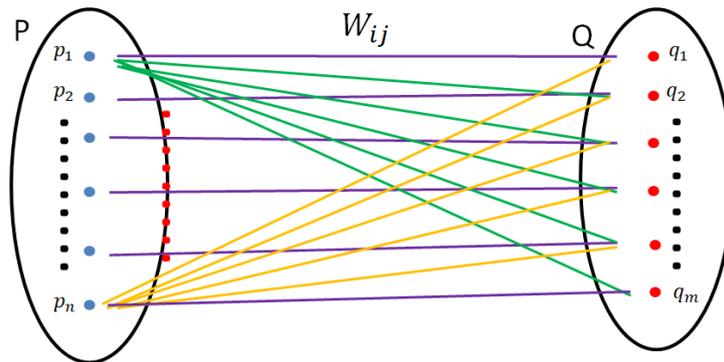


圖 4. 將兩個音樂片段作成完全二分圖

(四) 評估

JKUPDD 包含了五位音樂家的片段(pattern)與各個片段(pattern)的事件(occurrence)。

1. 計算相同片段(pattern)中的相似度

在五位音樂家的 `ground truth pattern`，相同片段(pattern)中的第一個與第二個事件(occurrence)進行計算，比較兩個事件(occurrence)的相似度是否接近 1.0。

2. 計算不同片段(pattern)中的相似度

在五位音樂家的 `ground truth pattern` 中，計算相同音樂家、不同片段(pattern)的第一個事件(occurrence)，比較兩者的相似度是否足夠小。

3. 尋找重複片段

設定迭代次數 t 為 100，調整不同的鄰居數目 K 與緊密程度閾值(`cmp_thr`)，以找出對每個音樂家最適合的組合。

參、研究結果與討論

一、相同 pattern

(一) Bach

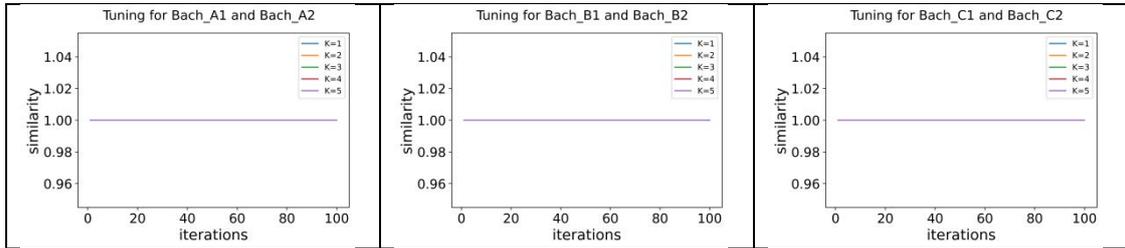


圖 5. 至圖 7. Bach 相同片段的事件一、二相似度隨迭代次數變化

(二) Beethoven

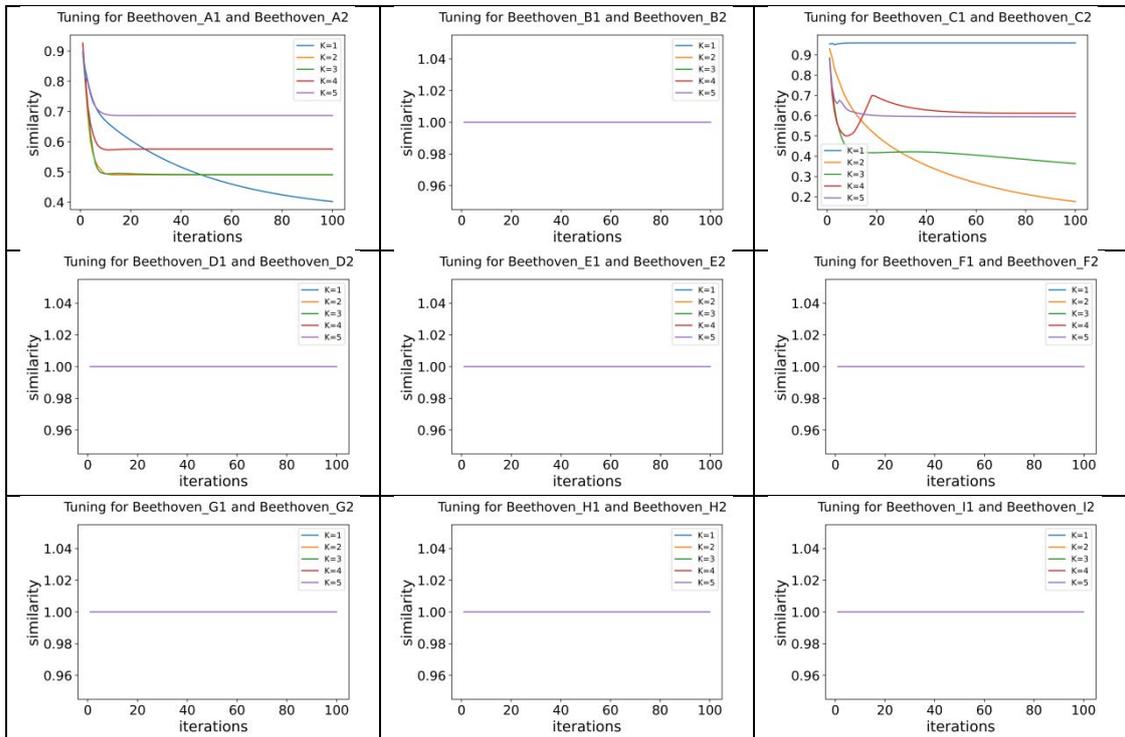


圖 8. 至圖 16. Beethoven 相同片段的事件一、二相似度隨迭代次數變化

(三) Chopin

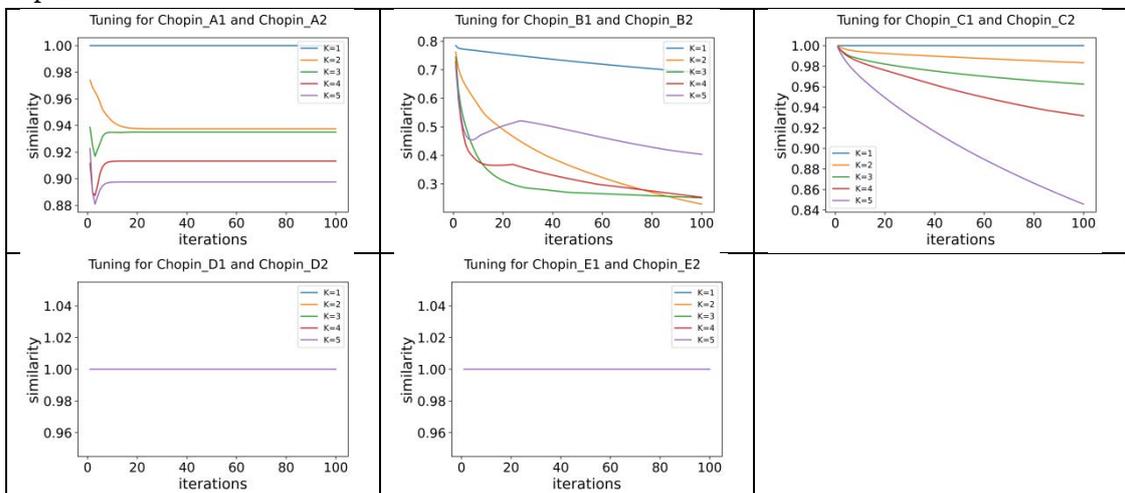


圖 17. 至圖 21. Chopin 相同片段的事件一、二相似度隨迭代次數變化

(四)Gibbons

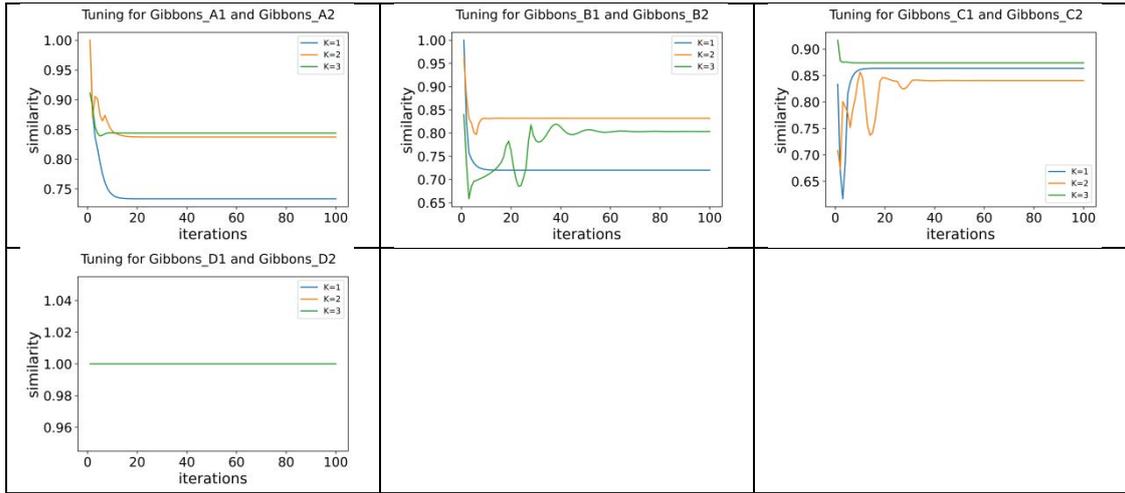


圖 22. 至 25. Gibbons 相同片段的事件一、二相似度隨迭代次數變化

(五)Mozart

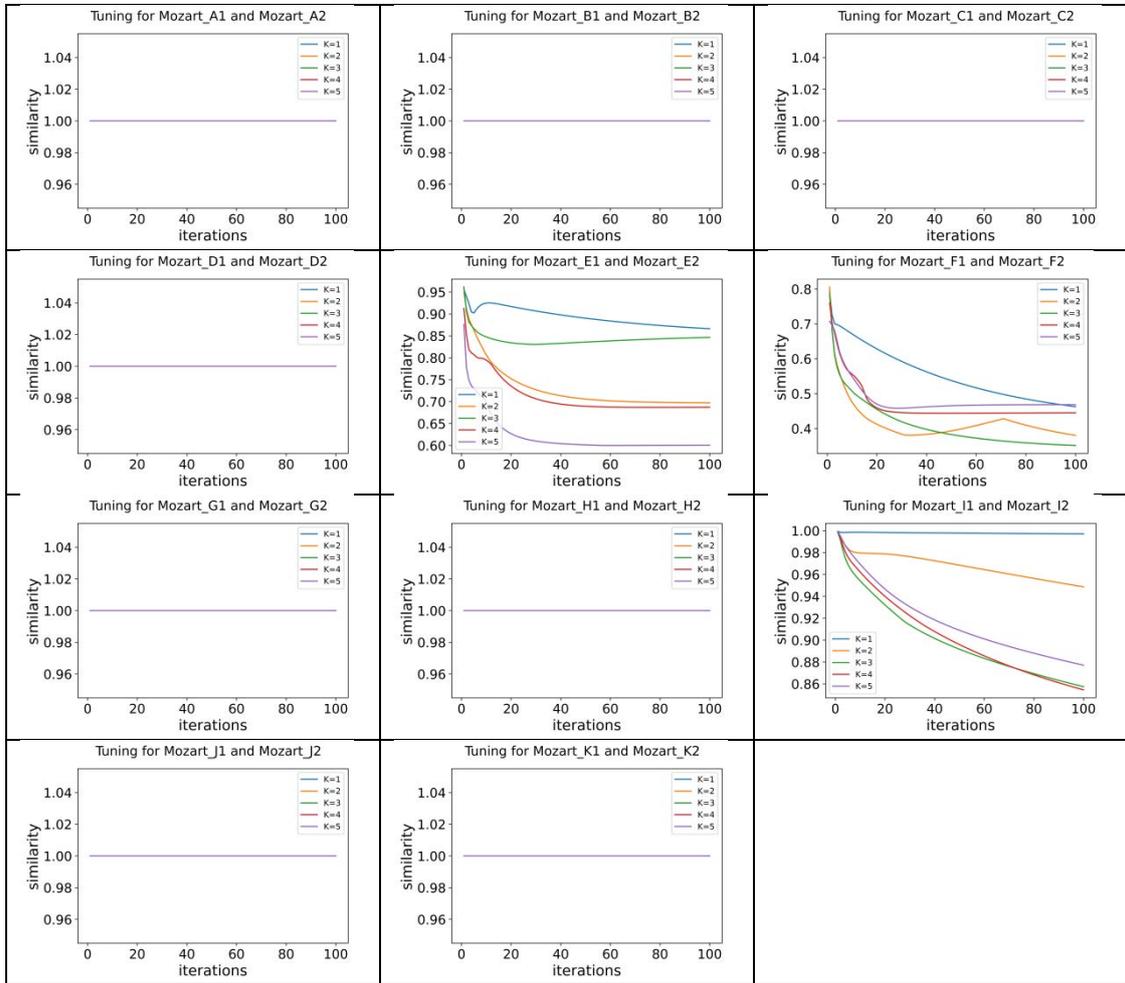


圖 26. 至圖 36. Mozart 相同片段的事件一、二相似度隨迭代次數變化

二、相異 pattern

(一) Bach

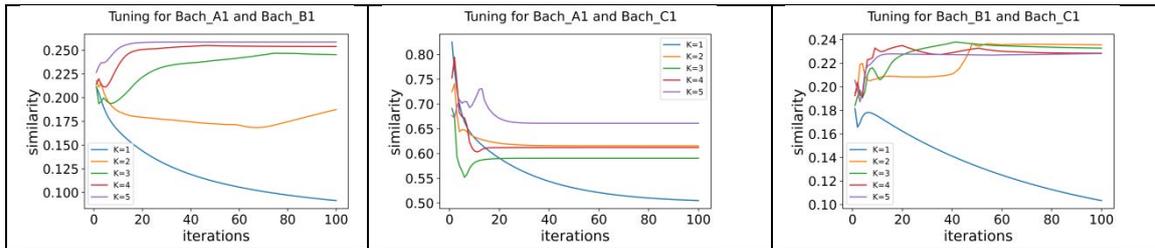
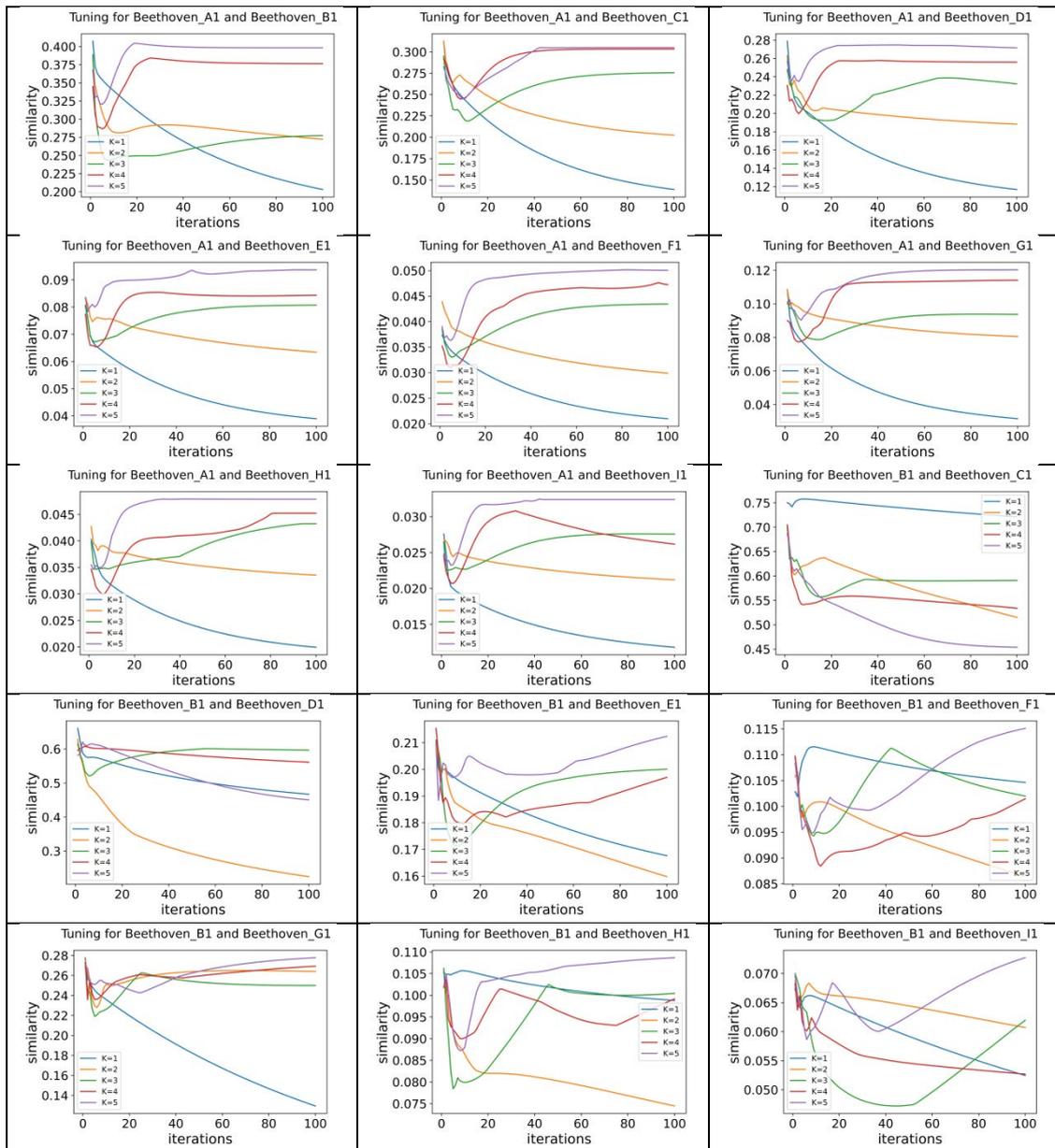


圖 37. 至圖 39. Bach 相異片段的事件一相似度隨迭代次數變化

(二) Beethoven



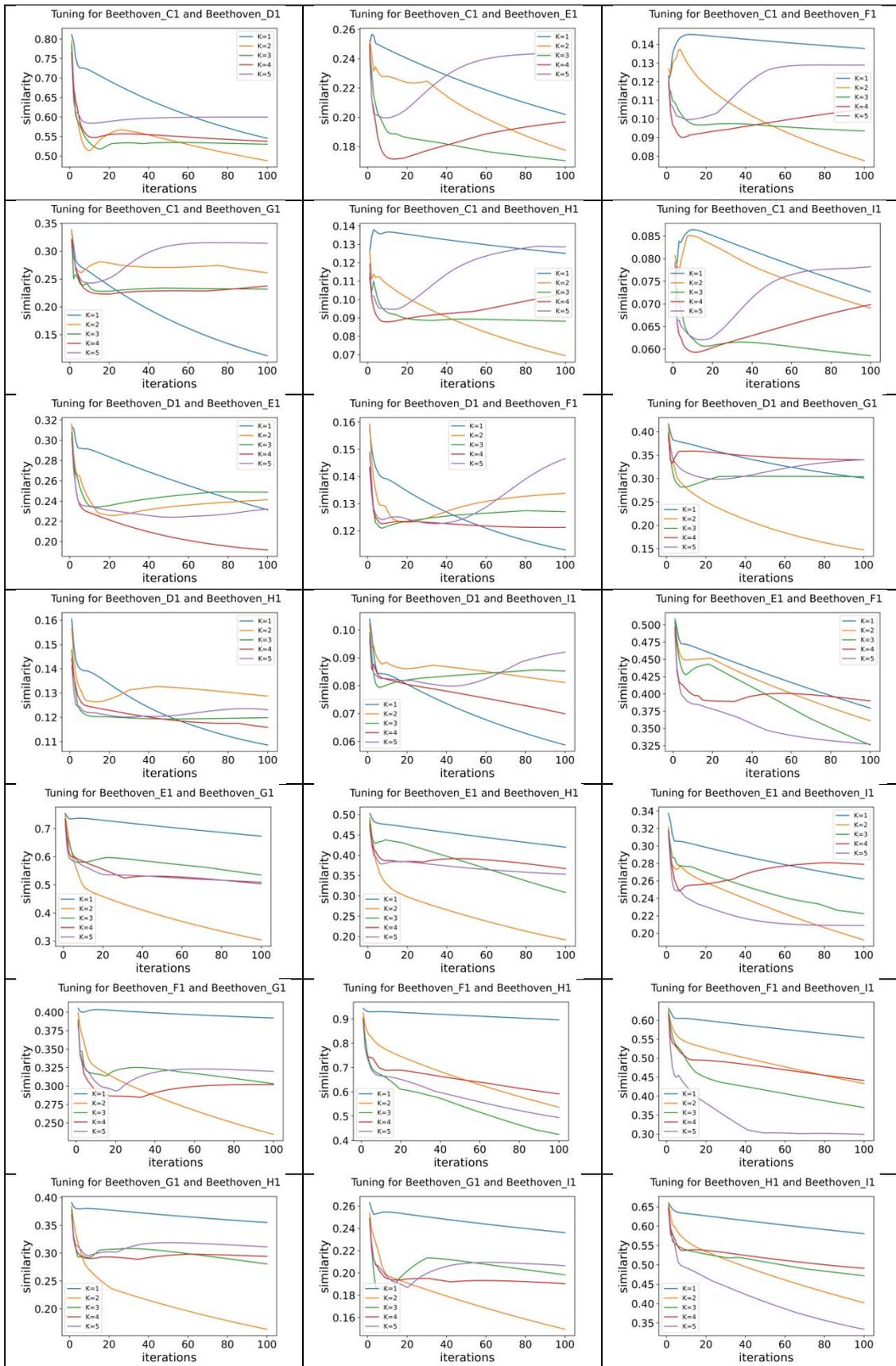


圖 40. 至圖 75. Beethoven 相異片段的事件一相似度隨迭代次數變化

(三)Chopin

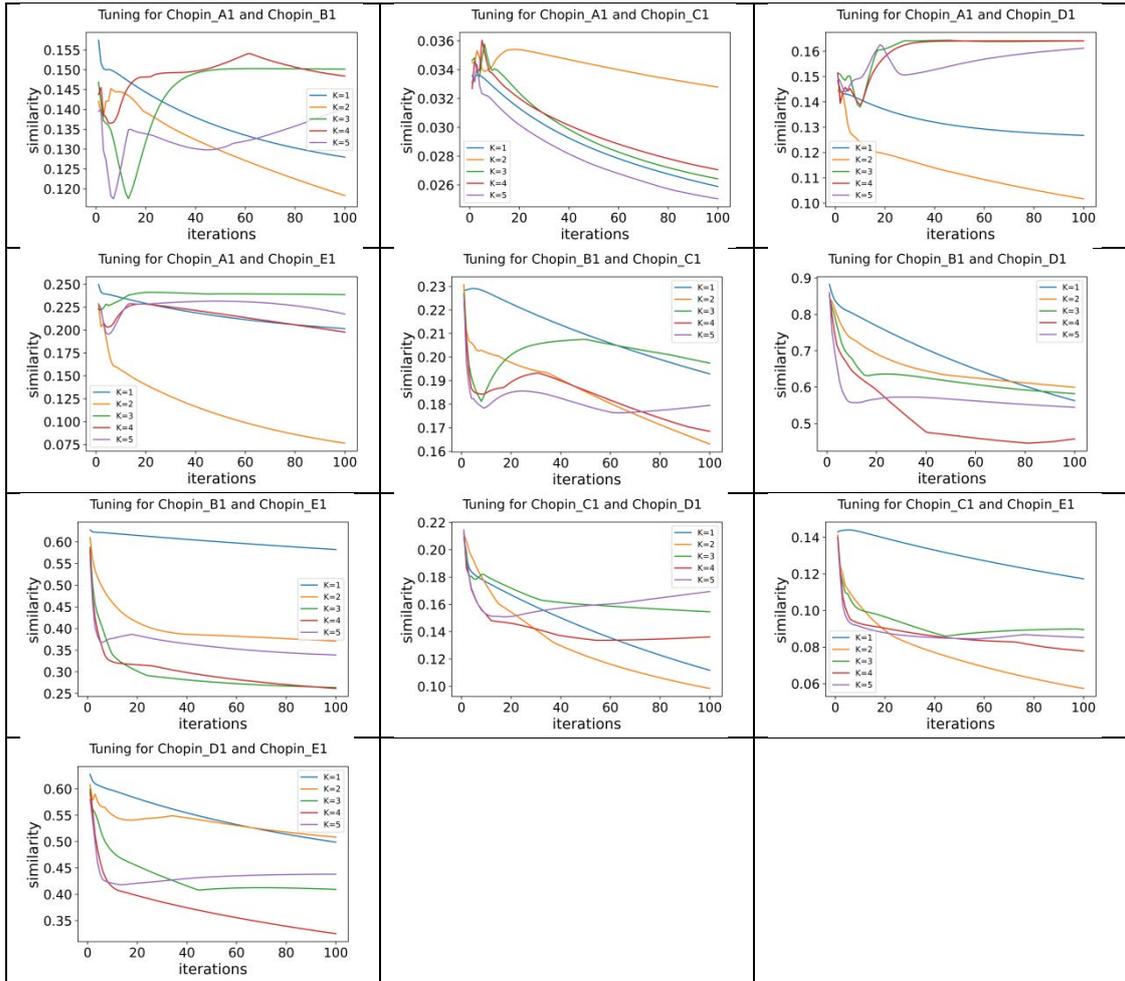


圖 76. 至圖 85. Chopin 相異片段的事件一相似度隨迭代次數變化

(四)Gibbons

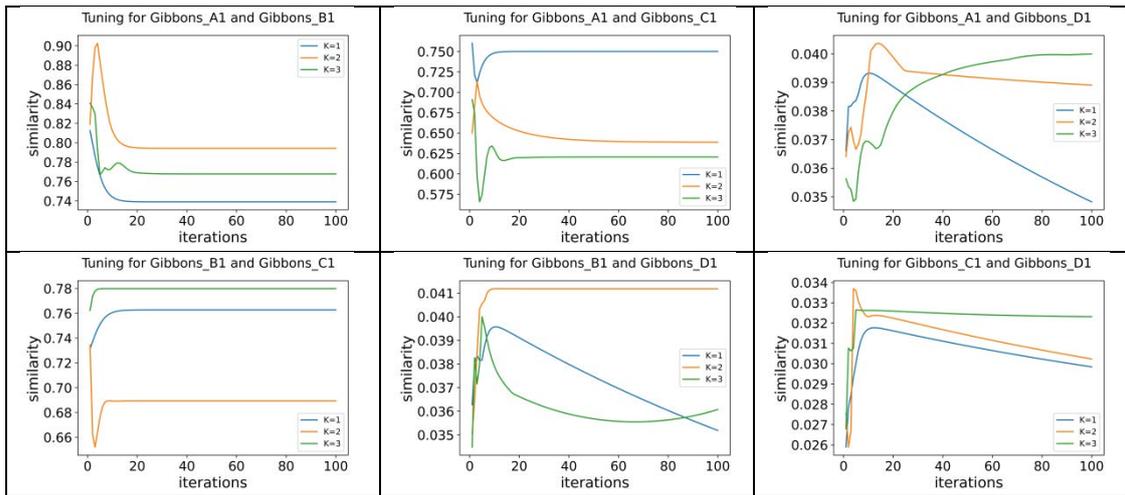
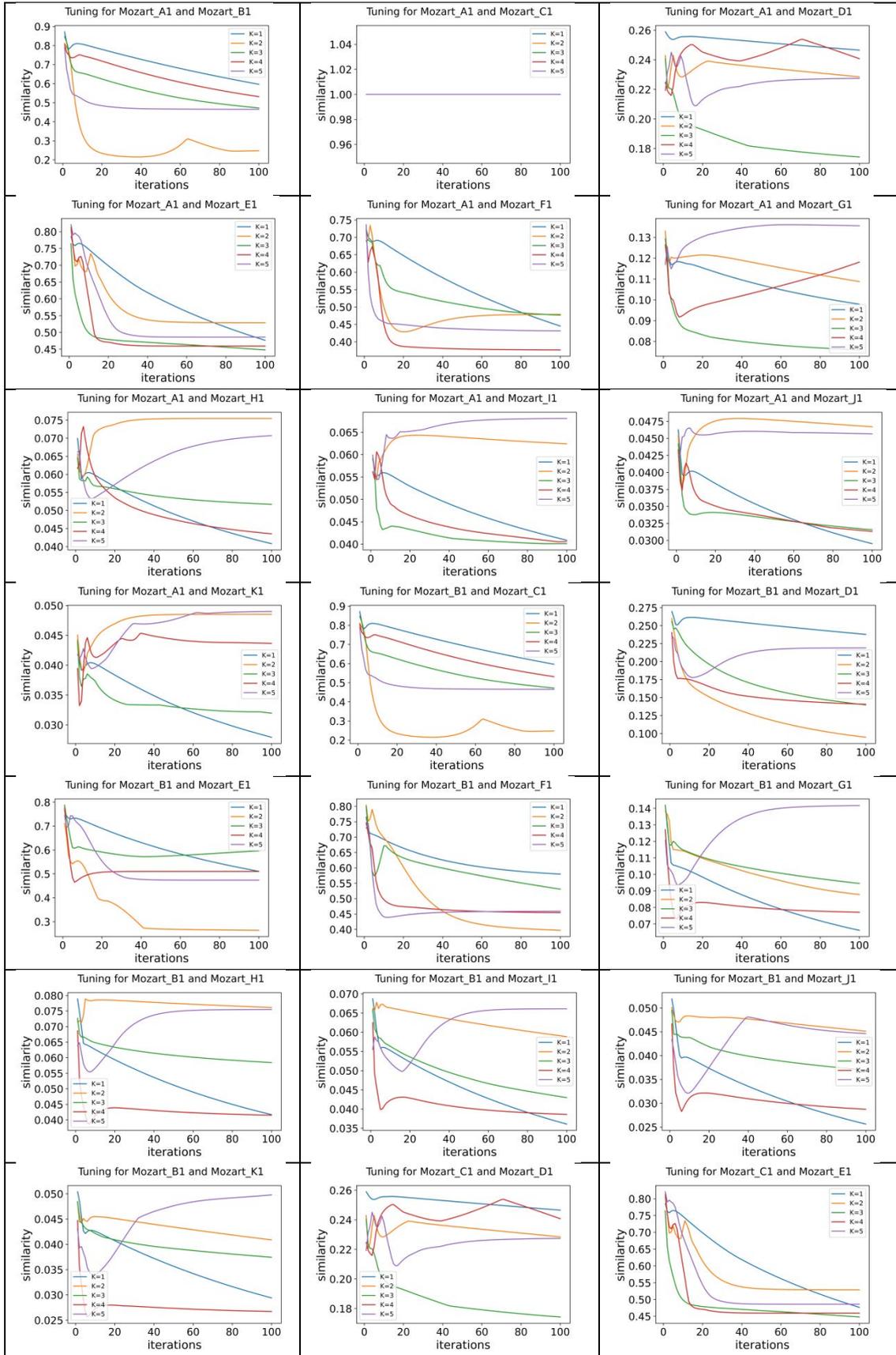
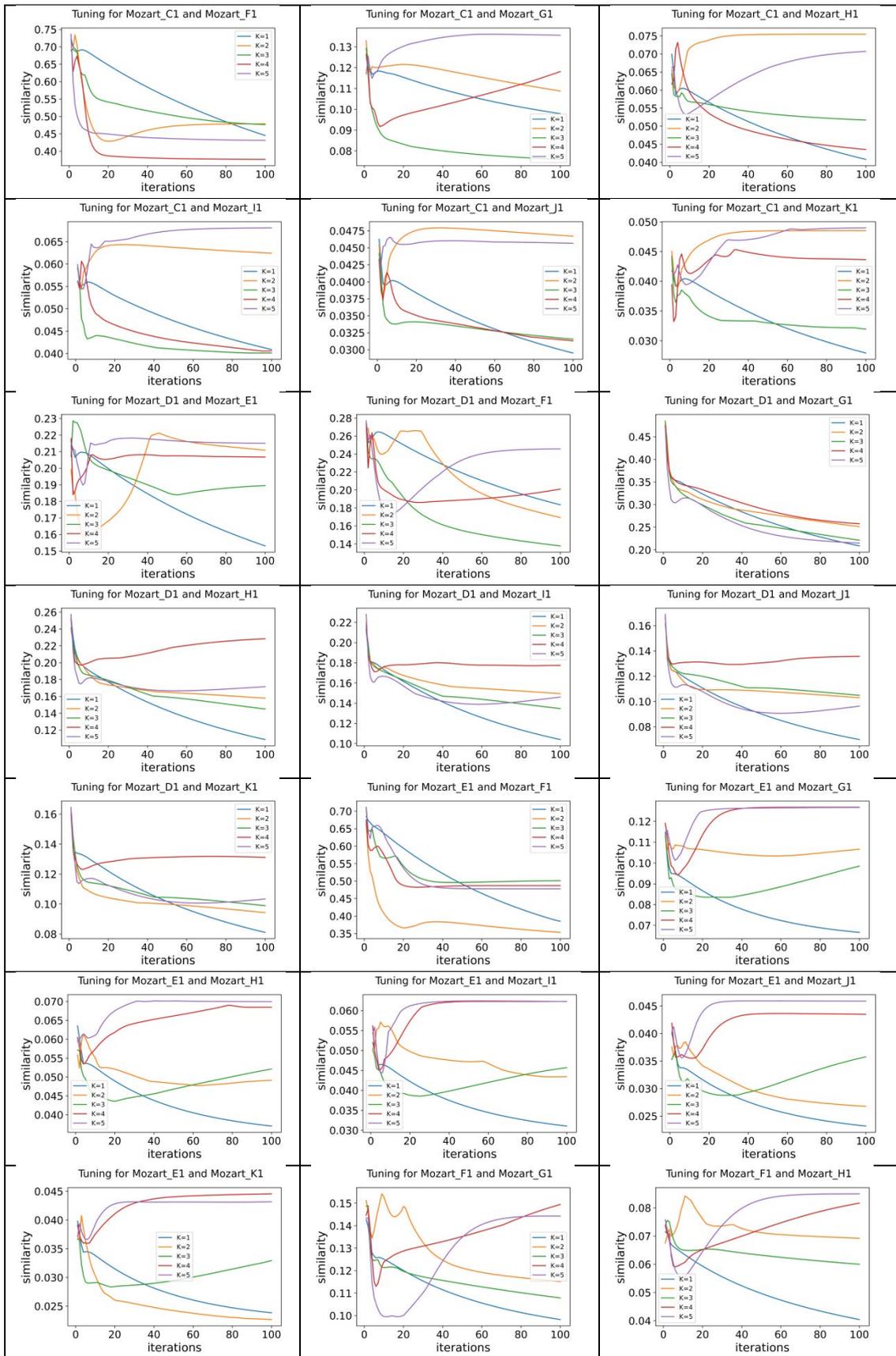


圖 86. 至圖 91. Gibbons 相異片段的事件一相似度隨迭代次數變化

(五)Mozart





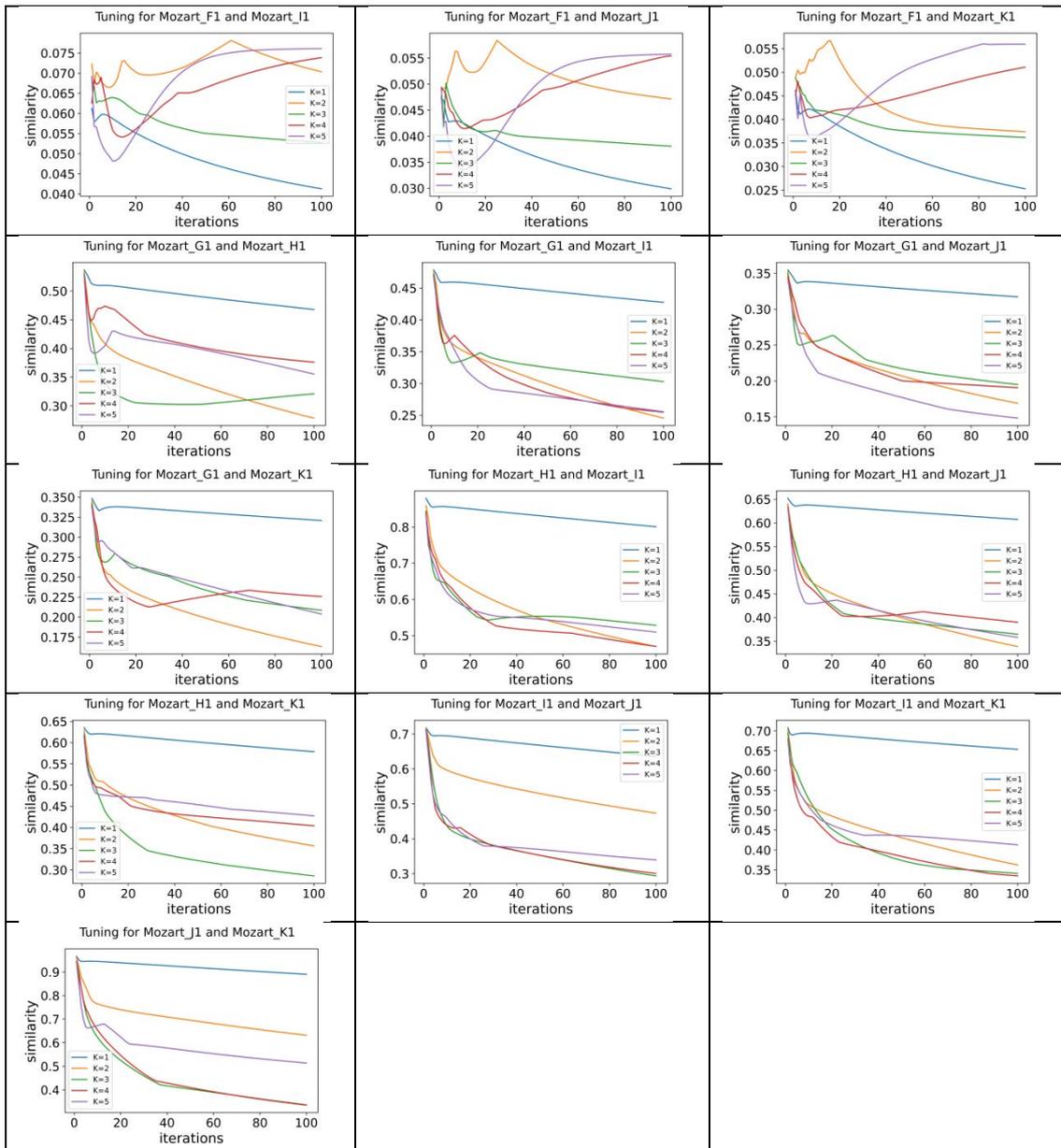


圖 92. 至圖 146. Mozart 相異片段的事件一相似度隨迭代次數變化

三、重複片段

將各個音樂家的整首樂曲視為一個音樂片段 P，將 P 與自身作成二分圖計算權重 W，印出自相似矩陣(self-similarity matrix)，可以觀察到顏色較深的小矩形、或是平行對角線的斜直線為重複片段處。

(一) Bach

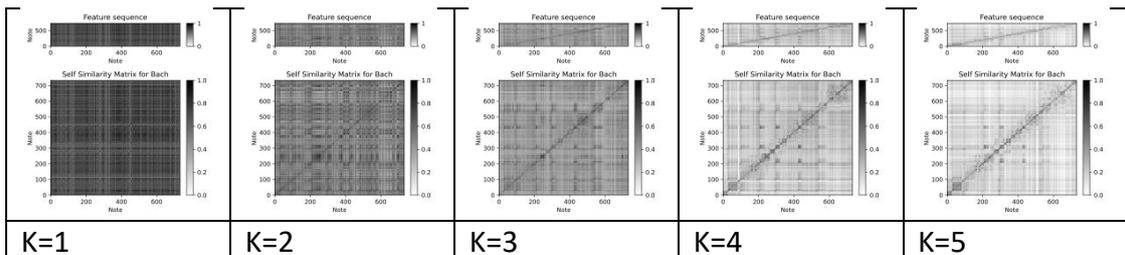


圖 147. 至圖 151. Bach 自相似矩陣(self-similarity matrix)隨鄰居數量 K 的變化

(二)Beethoven

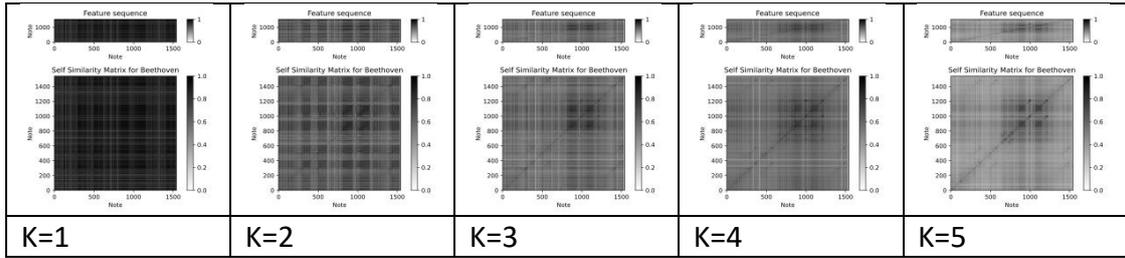


圖 152. 至圖 156. Beethoven 自相似矩陣(self-similarity matrix)隨鄰居數量 K 的變化

(三)Chopin

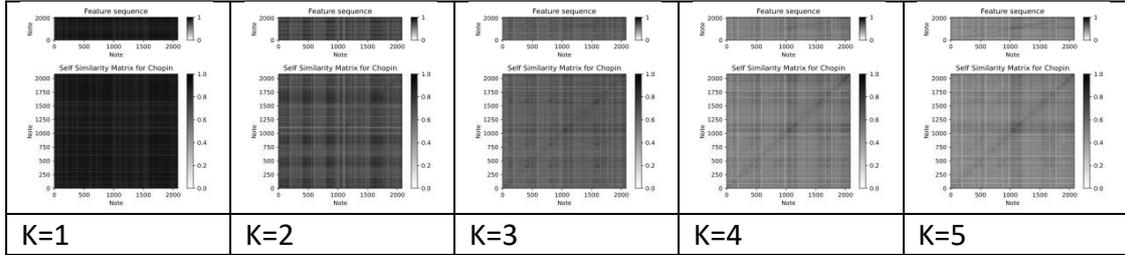


圖 157. 至圖 161. Chopin 自相似矩陣(self-similarity matrix)隨鄰居數量 K 的變化

(四)Gibbons

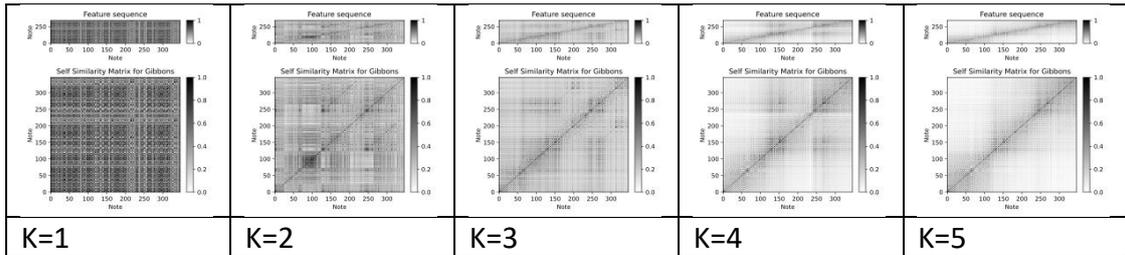


圖 162. 至圖 166. Gibbons 自相似矩陣(self-similarity matrix)隨鄰居數量 K 的變化

(五)Mozart

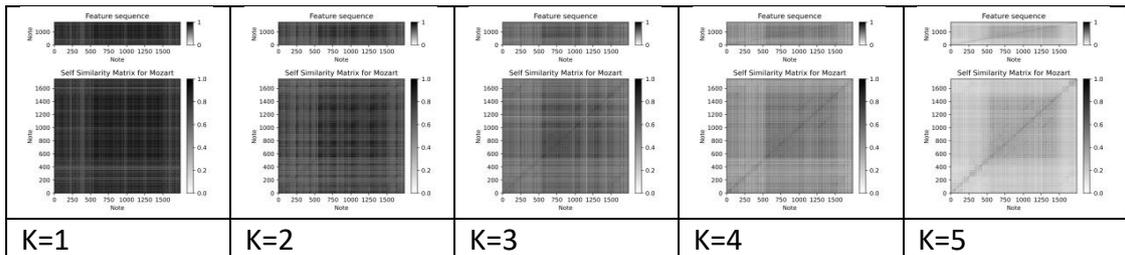


圖 167. 至圖 171. Mozart 自相似矩陣(self-similarity matrix)隨鄰居數量 K 的變化

四、尋找相似片段

我們對 JKUPDD 中的五位音樂家的樂曲，嘗試了鄰居數量 $K=1\sim 5$ 、緊密程度閾值(cmp_thr)= $0、0.5、1、1.5、2$ ，以下列出在 standard, establishment, occurrence, three laryer 的表現。

(一)Bach

表 1. Bach 樂曲在不同鄰居數量與緊密程度閾值組合的表現

Bach	F1_std	P_std	R_std	F1_est	P_est	R_est	F1_occ	P_occ	R_occ	F1_thr	P_thr	R_thr
(1,0)	0.00000	0.00000	0.00000	0.21062	0.20331	0.21847	0.00000	0.00000	0.00000	0.09259	0.11339	0.07824
(1,0.5)	0.00000	0.00000	0.00000	0.08454	0.09546	0.07586	0.00000	0.00000	0.00000	0.06117	0.07797	0.05033
(1,1)	0.00000	0.00000	0.00000	0.08806	0.10493	0.07586	0.00000	0.00000	0.00000	0.05887	0.07115	0.05021
(1,1.5)	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
(1,2)												
(2,0)	0.00000	0.00000	0.00000	0.39991	0.42434	0.37813	0.00000	0.00000	0.00000	0.17416	0.23678	0.13773
(2,0.5)	0.00000	0.00000	0.00000	0.06250	0.12500	0.04167	0.00000	0.00000	0.00000	0.01333	0.02667	0.00889
(2,1)	0.00000	0.00000	0.00000	0.06250	0.12500	0.04167	0.00000	0.00000	0.00000	0.01333	0.02667	0.00889
(2,1.5)	0.00000	0.00000	0.00000	0.06250	0.12500	0.04167	0.00000	0.00000	0.00000	0.01333	0.02667	0.00889
(2,2)	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
(3,0)	0.00000	0.00000	0.00000	0.23808	0.37030	0.17544	0.00000	0.00000	0.00000	0.11930	0.17663	0.09007
(3,0.5)	0.00000	0.00000	0.00000	0.08571	0.10714	0.07143	0.00000	0.00000	0.00000	0.04344	0.05430	0.03620
(3,1)	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
(3,1.5)	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
(3,2)	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
(4,0)	0.00000	0.00000	0.00000	0.17103	0.30357	0.11905	0.00000	0.00000	0.00000	0.11333	0.20733	0.07797
(4,0.5)	0.00000	0.00000	0.00000	0.16807	0.28571	0.11905	0.00000	0.00000	0.00000	0.10639	0.16738	0.07797
(4,1)	0.00000	0.00000	0.00000	0.16807	0.28571	0.11905	0.00000	0.00000	0.00000	0.10716	0.17126	0.07797
(4,1.5)	0.00000	0.00000	0.00000	0.10714	0.21429	0.07143	0.00000	0.00000	0.00000	0.03209	0.06417	0.02139
(4,2)												
(5,0)	0.00000	0.00000	0.00000	0.22897	0.35714	0.16850	0.00000	0.00000	0.00000	0.14373	0.23344	0.10383
(5,0.5)	0.00000	0.00000	0.00000	0.12500	0.25000	0.08333	0.00000	0.00000	0.00000	0.05792	0.11584	0.03861
(5,1)	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
(5,1.5)	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
(5,2)	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000

(二)Beethoven

表 2. Beethoven 樂曲在不同鄰居數量與緊密程度閾值組合的表現

Beethoven	F1_std	P_std	R_std	F1_est	P_est	R_est	F1_occ	P_occ	R_occ	F1_thr	P_thr	R_thr
(1,0)	0.00000	0.00000	0.00000	0.40979	0.37833	0.44697	0.00000	0.00000	0.00000	0.28206	0.27356	0.29111
(1,0.5)	0.00000	0.00000	0.00000	0.42550	0.41157	0.44041	0.00000	0.00000	0.00000	0.30096	0.31151	0.29111
(1,1)	0.00000	0.00000	0.00000	0.31125	0.28529	0.34241	0.00000	0.00000	0.00000	0.28355	0.35645	0.23541
(1,1.5)	0.00000	0.00000	0.00000	0.22072	0.30392	0.17329	0.00000	0.00000	0.00000	0.26073	0.39478	0.19463
(1,2)												
(2,0)	0.00000	0.00000	0.00000	0.22321	0.19420	0.26242	0.00000	0.00000	0.00000	0.19655	0.16810	0.23660
(2,0.5)	0.00000	0.00000	0.00000	0.22267	0.21548	0.23035	0.00000	0.00000	0.00000	0.19081	0.18363	0.19857
(2,1)	0.00000	0.00000	0.00000	0.22267	0.21548	0.23035	0.00000	0.00000	0.00000	0.19330	0.18629	0.20085
(2,1.5)	0.00000	0.00000	0.00000	0.01913	0.03431	0.01327	0.00000	0.00000	0.00000	0.03126	0.05600	0.02168
(2,2)	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
(3,0)	0.00000	0.00000	0.00000	0.07689	0.08944	0.06743	0.00000	0.00000	0.00000	0.07325	0.09068	0.06144
(3,0.5)	0.00000	0.00000	0.00000	0.07689	0.08944	0.06743	0.00000	0.00000	0.00000	0.07631	0.09612	0.06327
(3,1)	0.00000	0.00000	0.00000	0.08248	0.10619	0.06743	0.00000	0.00000	0.00000	0.06293	0.08478	0.05003
(3,1.5)	0.00000	0.00000	0.00000	0.07556	0.12803	0.05360	0.00000	0.00000	0.00000	0.06456	0.11875	0.04433
(3,2)	0.00000	0.00000	0.00000	0.01596	0.02941	0.01096	0.00000	0.00000	0.00000	0.01450	0.02740	0.00986
(4,0)	0.00000	0.00000	0.00000	0.07569	0.14216	0.05157	0.00000	0.00000	0.00000	0.09743	0.19557	0.06487
(4,0.5)	0.00000	0.00000	0.00000	0.06794	0.14216	0.04464	0.00000	0.00000	0.00000	0.09964	0.20789	0.06552
(4,1)	0.00000	0.00000	0.00000	0.06694	0.13990	0.04400	0.00000	0.00000	0.00000	0.07645	0.15694	0.05053
(4,1.5)	0.00000	0.00000	0.00000	0.06694	0.13990	0.04400	0.00000	0.00000	0.00000	0.07011	0.14015	0.04675
(4,2)	0.00000	0.00000	0.00000	0.00704	0.02817	0.00402	0.00000	0.00000	0.00000	0.01370	0.05479	0.00783
(5,0)	0.00000	0.00000	0.00000	0.13036	0.14005	0.12192	0.00000	0.00000	0.00000	0.13786	0.15953	0.12137
(5,0.5)	0.00000	0.00000	0.00000	0.13494	0.16022	0.11656	0.00000	0.00000	0.00000	0.14781	0.18951	0.12115
(5,1)	0.00000	0.00000	0.00000	0.13494	0.16022	0.11656	0.00000	0.00000	0.00000	0.14971	0.18766	0.12452
(5,1.5)	0.00000	0.00000	0.00000	0.08391	0.12362	0.06350	0.00000	0.00000	0.00000	0.10162	0.15340	0.07597
(5,2)	0.00000	0.00000	0.00000	0.03820	0.07051	0.02620	0.00000	0.00000	0.00000	0.06972	0.12956	0.04770

(三)Chopin

表 3. Chopin 樂曲在不同鄰居數量與緊密程度閾值組合的表現

Chopin	F1 std	P std	R std	F1 est	P est	R est	F1 occ	P occ	R occ	F1 thr	P thr	R thr
(1,0)	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
(1,0.5)	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
(1,1)	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
(1,1.5)	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
(1,2)												
(2,0)												
(2,0.5)												
(2,1)												
(2,1.5)												
(2,2)	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
(3,0)												
(3,0.5)												
(3,1)	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
(3,1.5)	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
(3,2)	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
(4,0)												
(4,0.5)												
(4,1)	0.00000	0.00000	0.00000	0.05728	0.08410	0.04342	0.00000	0.00000	0.00000	0.02207	0.03171	0.01692
(4,1.5)	0.00000	0.00000	0.00000	0.02731	0.05804	0.01786	0.00000	0.00000	0.00000	0.00996	0.02027	0.00660
(4,2)	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
(5,0)												
(5,0.5)												
(5,1)	0.00000	0.00000	0.00000	0.06499	0.09094	0.05056	0.00000	0.00000	0.00000	0.06003	0.08407	0.04668
(5,1.5)	0.00000	0.00000	0.00000	0.06349	0.09677	0.04724	0.00000	0.00000	0.00000	0.03885	0.06559	0.02759
(5,2)	0.00000	0.00000	0.00000	0.01786	0.04464	0.01116	0.00000	0.00000	0.00000	0.00678	0.01695	0.00424

(四)Gibbons

表 4. Gibbons 樂曲在不同鄰居數量與緊密程度閾值組合的表現

Gibbons	F1 std	P std	R std	F1 est	P est	R est	F1 occ	P occ	R occ	F1 thr	P thr	R thr
(1,0)	0.00000	0.00000	0.00000	0.09576	0.11442	0.08234	0.00000	0.00000	0.00000	0.11071	0.14799	0.08844
(1,0.5)	0.00000	0.00000	0.00000	0.09502	0.11230	0.08234	0.00000	0.00000	0.00000	0.11242	0.15426	0.08844
(1,1)	0.00000	0.00000	0.00000	0.03980	0.07738	0.02679	0.00000	0.00000	0.00000	0.07127	0.14160	0.04762
(1,1.5)	0.00000	0.00000	0.00000	0.02500	0.06250	0.01562	0.00000	0.00000	0.00000	0.04702	0.11716	0.02941
(1,2)	0.00000	0.00000	0.00000	0.02500	0.06250	0.01562	0.00000	0.00000	0.00000	0.04702	0.11716	0.02941
(2,0)	0.00000	0.00000	0.00000	0.12162	0.25000	0.08036	0.00000	0.00000	0.00000	0.10738	0.17582	0.07729
(2,0.5)	0.00000	0.00000	0.00000	0.12162	0.25000	0.08036	0.00000	0.00000	0.00000	0.10738	0.17582	0.07729
(2,1)	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
(2,1.5)	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
(2,2)												
(3,0)	0.00000	0.00000	0.00000	0.20388	0.28571	0.15848	0.00000	0.00000	0.00000	0.08065	0.10084	0.06720
(3,0.5)	0.00000	0.00000	0.00000	0.20388	0.28571	0.15848	0.00000	0.00000	0.00000	0.08065	0.10084	0.06720
(3,1)	0.00000	0.00000	0.00000	0.20388	0.28571	0.15848	0.00000	0.00000	0.00000	0.08065	0.10084	0.06720
(3,1.5)	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
(3,2)												
(4,0)	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
(4,0.5)	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
(4,1)	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
(4,1.5)												
(4,2)												
(5,0)												
(5,0.5)												
(5,1)												
(5,1.5)												
(5,2)												

(五)Mozart

表 5. Mozart 樂曲在不同鄰居數量與緊密程度閾值組合的表現

Gibbons	F1_std	P_std	R_std	F1_est	P_est	R_est	F1_occ	P_occ	R_occ	F1_thr	P_thr	R_thr
(1,0)	0.00000	0.00000	0.00000	0.09576	0.11442	0.08234	0.00000	0.00000	0.00000	0.11071	0.14799	0.08844
(1,0.5)	0.00000	0.00000	0.00000	0.09502	0.11230	0.08234	0.00000	0.00000	0.00000	0.11242	0.15426	0.08844
(1,1)	0.00000	0.00000	0.00000	0.03980	0.07738	0.02679	0.00000	0.00000	0.00000	0.07127	0.14160	0.04762
(1,1.5)	0.00000	0.00000	0.00000	0.02500	0.06250	0.01562	0.00000	0.00000	0.00000	0.04702	0.11716	0.02941
(1,2)	0.00000	0.00000	0.00000	0.02500	0.06250	0.01562	0.00000	0.00000	0.00000	0.04702	0.11716	0.02941
(2,0)	0.00000	0.00000	0.00000	0.12162	0.25000	0.08036	0.00000	0.00000	0.00000	0.10738	0.17582	0.07729
(2,0.5)	0.00000	0.00000	0.00000	0.12162	0.25000	0.08036	0.00000	0.00000	0.00000	0.10738	0.17582	0.07729
(2,1)	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
(2,1.5)	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
(2,2)												
(3,0)	0.00000	0.00000	0.00000	0.20388	0.28571	0.15848	0.00000	0.00000	0.00000	0.08065	0.10084	0.06720
(3,0.5)	0.00000	0.00000	0.00000	0.20388	0.28571	0.15848	0.00000	0.00000	0.00000	0.08065	0.10084	0.06720
(3,1)	0.00000	0.00000	0.00000	0.20388	0.28571	0.15848	0.00000	0.00000	0.00000	0.08065	0.10084	0.06720
(3,1.5)	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
(3,2)												
(4,0)	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
(4,0.5)	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
(4,1)	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
(4,1.5)												
(4,2)												
(5,0)												
(5,0.5)												
(5,1)												
(5,1.5)												
(5,2)												

五、討論

(一)相同音樂家的相同片段

JKUPDD 中相同音樂家的相同片段的第一、二個事件，有些情形得到了低的相似度值，有可能是因為第二個事件與第一個事件有一些變化，或是因為更新權重時，「按照最近的鄰居順序更新」造成。

(二)相同音樂家的相異片段

JKUPDD 中相同音樂家的相異片段的第一個事件，有些得到了高的相似度值，可能是因為更新權重時，「按照最近的鄰居順序更新」造成。

(三)自相似矩陣

自相似矩陣上可以看到 K=3、4、5 時，重複片段的現象較為明顯，但是結果卻是 K=1、2 時有較好的表現。可能是因為音符相似群在合併的時候，沒有考慮到前後若干個音符的平移問題。

(四)不同鄰居數量與緊密程度閾值組合的表現

不同音樂家擁有較好表現的組合不一樣，其中 Chopin 甚至沒有任何超過一項 0.2，推測可能是和「按照最近的鄰居順序更新」、音符相似群合併未考慮平移，或是和音樂家的樂曲有關。

肆、結論與應用

一、結論

(一)相似性度量的調整參數

1. 懲罰

以 $\min\{n, m\}$ 作為懲罰導致有些相異片段的相似度過高；以 $\max\{n, m\}$ 作為懲罰較為合理。

2. 鄰居數量

鄰居數量隨著其他參數的設定，對相似度的影響不同。鄰居數量 $K=1、2$ 的表現最好；鄰居數量 $K=3、4、5$ 則是使重複片段在自相似矩陣上較為明顯。

3. 迭代次數

考慮相同與相異片段(pattern)的計算，多數情況的相似度值在 $t=100$ 時已經收斂。

(二)重複片段觀察

1. 自相似矩陣中的重複片段，在計算權重時考慮了鄰居之間的權重，造成「重複片段」不是呈現一個音符與一個音符之間的高對應分數，而是多個音符與多個音符之間的高對應分數。

2. 當鄰居數量 $K=3\sim 5$ 時，重複片段較明顯，可以看到若干平行對角線的斜直線。

(三)資料集上的測試

當鄰居數量 $K=1、2$ ，緊密程度閾值 $=0\sim 1.5$ 時，有較好的表現。其中 Chopin 完全沒有超過 0.2 的項目。

二、應用

(一)音樂資訊檢索(music information retrieval)

音樂相似性度量可以透過計算兩段音樂的相似度，進行音樂的搜尋，或是音樂推薦系統(music recommender system)等等。

(二)音樂生成(music generation)

音樂相似性度量可以透過比較兩段音樂的相似程度，選取較好的訓練資料，可以應用於提升音樂風格轉換(music style transfer)的表現。

伍、參考文獻

- [1] Meredith, D., Lemström, K., & Wiggins, G. A. (2002). Algorithms for discovering repeated patterns in multidimensional representations of polyphonic music. *Journal of New Music Research*, 31(4), 321-345.
- [2] Meredith, D. (2006). Point-set algorithms for pattern discovery and pattern matching in music. In *Dagstuhl Seminar Proceedings*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik.
- [3] Chen, T. P., & Su, L. (2017). Discovery of repeated themes and sections with pattern clustering. *Music Information Retrieval Evaluation eXchange (MIREX 2017)*.
- [4] Chen, T. P., & Su, L. (2018, November). The Musical Schemagram: Time-scale Visualization of Repeated Patterns in Music. In *2018 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)* (pp. 1642-1648). IEEE.
- [5] MIREX 2017: Discovery of Repeated Themes & Sections. https://www.music-ir.org/mirex/wiki/2017:Discovery_of_Repeated_Themes_%26_Sections
- [6] Liu, X., Han, C., & Guo, T. (2015, June). A robust point sets matching method. In *International Conference in Swarm Intelligence* (pp. 383-390). Springer, Cham.

【評語】 190005

這是有趣的研究題目，作者提出一種對音樂片段相似性的度量且探討要如何調整其所用的參數值，但缺乏相關研究的比較和探討。此外，此作品在比較相似度時只有使用一些數值指標但沒有讓真人來評估相似度，這部分可以加強。