クロマ類似度を用いた音楽データベースの圧縮

Compression of musical audio database with chroma's similarity measure

橋口 博樹　工藤 雅志

Hiroki Hashiguchi and Masashi Kudo

We have developed a music retrieval system that takes a humming query and finds similar audio intervals (segments) in a musical audio database. This system enables a user to retrieve a segment of a musical audio signal desired just by singing its melody. In this paper, we propose a method to compress the music database through similarity analysis. The distance of chroma vectors is used as a similarity measure. The aim of this compression is to reduce the retrieval time. Practical experiments was conducted by actually using 115 musical audios in the RWC popular music database. We report that the compression ratio is about 45 %.

Keyword: Chroma vector, Music information retrieval, Pattern discovery, Moving average, Discriminant analysis.

1　はじめに

近年，携帯型音楽プレーヤーや音楽のインターネット経由での配信など，音楽情報の産業応用が急激に拡大し，大量の楽曲に対する計算機での容易な処理が課題となっている[8]．このような状況の中，著者らは，検索という観点から，楽曲の音響信号をデータベース，鼻歌をクエリ（検索要求）とした鼻歌検索 (ハミング検索) に着手し，検索手法の提案[11][10][9] を行っている．一方，要約の観点から．楽曲中で繰り返される類似区間 (フレーズなど) の検出するための類似性解析に基づいて，楽曲の代表的な部分 (多くの場合サビ) を一箇所切り出す手法[1, 2] や，主要な部分を残して短くする音楽要約手法[3, 13]，全てのサビ区間を網羅的に検出する手法 (Refrain Detecting Method:RefraiD)[6] などが提案されている．

本稿では，人間が曲をあらかじめ聴いて曲構成を作り，それに対してコンピュータが類似性を判断して推定した曲構成を比較する．その際，類似性の解析には RefraiD の考え方を取り入れ，楽曲の構成を要約する．さらに，鼻歌から楽曲を検索する際の検索時間の短縮を目的として，要約を基に楽曲データベースの圧縮を行う．

*埼玉大学工学部情報システム工学科

Department of Information and Computer Sciences, Faculty of Engineering, Saitama University, 255 Shimo-Okubo, Sakura-ku, Saitama, Saitama, 338-8570, Japan

2　楽曲信号の圧縮

2.1　圧縮の定義

典型的なポピュラー音楽の楽曲は，前奏 (イントロ) → 1A メロ → 1B メロ → 1 サビ　→ 2A メロ → 2B メロ → 2 サビ → 間奏→ 3B メロ → 3 サビ→ 4 サビ → 後奏 (アウトロ) のような構成をしている．この中のメロディが類似している区間，つまり繰り返し区間を検出し，間奏や後奏を除いた繰り返し区間のみで楽曲を再構成する．「前奏 (イントロ) → A メロ → B メロ → サビ」のように再構成を行う要約を楽曲信号の圧縮と呼ぶ．

2.2　クロマベクトルとクロマ類似度

音階の基本周波数は平均律に従うものとする．本実装では，A1 (55Hz) から A8 (7040Hz) の帯域を扱い，各音の高さ (音高) を $C = \{1, \ldots, 12 \times 7\}$ の要素で表す．ここに，音高 c は $\omega(c) = 55 \cdot 2^{(c-1)/12}$[Hz] の周波数に対応する．音楽音響信号を標本化周波数 16kHz，量子化ビット

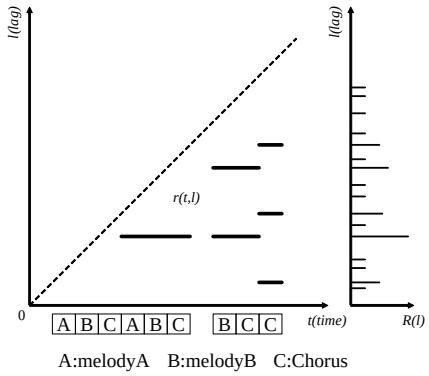

Fig. 1: Illustration of chroma's similarity measure: $r(t,l)$

数 16 ビットで A-D 変換し，モノラル録音する．FFT では，窓幅を 2048 点，シフト幅を 1024 点とし，1 フレームの単位時間は 64ms，1 秒間に約 16 個の音高を抽出する．フレーム t, 周波数 $\omega(c)$ のパワースペクトルを $f(w(c),t)$ と表す．また曲の長さ（フレームの総数）を W で表す．

平均律の異なる音名 $c(1 \leq c \leq 12)$ の加算されたパワー $v_c(t)$ を

$$v_c(t) = \sum_{h=1}^{7} f(\omega(12(h-1)+c), t) \tag{1}$$

と定義し，$\vec{v}(t) = (v_1(t), ..., v_{12}(t))$ をクロマベクトルと呼ぶ [6]．

フレーム t のクロマベクトル $\vec{v}(t) = (v_1(t), ..., v_{12}(t))$ と，それよりラグ (lag)$l(0 \leq l < t)$ だけ過去の $\vec{v}(t-l)$ とのクロマ類似度 $r(t,l)$ を，

$$r(t,l) = 1 - \frac{1}{\sqrt{12}} \left| \frac{\vec{v}(t)}{\max_c v_c(t)} - \frac{\vec{v}(t-l)}{\max_c v_c(t-l)} \right| \tag{2}$$

と定義する [6]．ここで，ベクトルのノルム $|\cdot|$ は通常のユークリッドノルムとする．分母の $\sqrt{12}$ は，1 辺の長さが 1 の 12 次元超立方体の対角線の長さに対応し，$r(t,l)$ を $0 \leq r(t,l) \leq 1$ に正規化している．クロマ類似度 $r(t,l)$ を，横軸に時間軸 t，縦軸がラグ軸 l の $t-l$ 平面に描画すると，繰り返されている区間に対応して，時間軸に平行な線分 (クロマ類似度が連続して高い領域) が Fig. 1 のように右下半分の三角形領域に現れる．Fig. 2 の左には，ある楽曲信号のクロマ類似度の一部を示す．後藤 [6] でも述べられているように $r(t,l)$ には，時間軸に垂直（上下），あるいは斜め右上・左下方向にノイズが現れ，これを後藤 [6] の方法でノイズ処理し，それを $r_d(t,l)$ と書く．Fig. 2 の中央には左図のノイズ処理後の $r_d(t,l)$ を示す．

Fig. 1 のように $t-l$ 平面で $r_d(t,l)$ を表現するので，

$$\{(t, l, r_d(t,l)) \,|\, t \in [t_1, t_2] \subset [0, W], r_d(t,l) \neq 0\}$$

(Left side) $r(l,t)$: chroma's similarity measure in the part of a real musical audio
(Center) $r_d(t,l)$: Noise reduction of $r(l,t)$ above
(Right side) $L(l)$: Lines corresponding to a similar segment

Fig. 2: Detection of similar segments

を l を固定したときの線分，あるいは単に線分という．

2.3　繰り返し区間の検出

Fig. 2 の中央に示すように至るところすべてに線分が存在する．しかし，抽出したい線分は $r_d(t,l)$ の値の大きい線分（類似線分）である．以下で類似線分の存在する可能性が高いと思われる l を検出する方法を述べる．時間軸と平行な一定区間 $(2Z+1$ フレーム) から，ラグ l におけるピーク値 $R(l)$ を (3) 式で求める (Fig. 1 右部分)．

$$R(l) = \sup_{l \leq t \leq W} \int_{t-Z_0}^{t+Z_0} \frac{r_d(\tau, l)}{2Z_0} d\tau \tag{3}$$

$$\approx \max_{l \leq t \leq W} \frac{1}{2Z_0+1} \sum_{\tau=\max\{0, t-Z_0\}}^{\min\{t+Z_0, W\}} r_d(\tau, l) \tag{4}$$

さらに，ノイズ成分の蓄積などによる大局的な変動を取り除くために，ピーク値 $R(l)$ のラグ軸方向で前後 Z_1 フレームの平均を取り，$R(l)$ から引く．この処理は，$R(l)$ にハイパスフィルタをかけることに相当する．

$$R'(l) = \max\left\{0, R(l) - \int_{l-Z_1}^{l+Z_1} \frac{R(\xi)}{2Z_1} d\xi\right\} \tag{5}$$

$$\approx \max\left\{0, R(l) - \frac{1}{2Z_1+1} \sum_{\xi=\max\{0, l-Z_1\}}^{\min\{l+Z_1, W\}} R(\xi)\right\}$$

$\{R'(1), \ldots, R'(W)\}$ の分布を調べ，類似線分がある・なしの判別を行う閾値 α を，自動閾値選定法 [12] を用いて決定する．この自動閾値選定法は，$\{R'(1), \ldots, R'(W)\}$ を 2 つのクラスに分けるときに，クラス分離度を最大とする判別基準である．

次に，$R'(l) > \alpha$ を満たす l において，クロマ類似度 $r_d(t,l)$ の時間軸 t 方向に線分の強調処理を行う．$R'(l) > \alpha$ を満たす l に対して，前後 Z_2 フレームの移動平均によっ

て平滑化した

$$r_s(t,l) = \int_{t-Z_2}^{t+Z_2} \frac{r_d(\tau,l)}{2Z_2} d\tau \approx \frac{1}{2Z_2+1} \sum_{\tau=t-Z_2}^{t+Z_2} r_d(\tau,l)$$

を求める．逆に $R'(l) \leq \alpha$ となる l については，$r_s(t,l) = 0$ とする．新たに $r_s(t,l)$ が閾値 β を連続して超える区間のうち，一定の長さ (Z_3) 以上の区間 $L(l)$ を (6) 式で定義し，類似区間とする．ここで，閾値 β は，α と同様に自動閾値選定法によって求める．

$$L(l) = \{[t_1,t_2] \,|\, r_s(t,l) > \beta \text{ for } \forall t \in [t_1,t_2], t_2 - t_1 > Z_3, 0 \leq t_1 < t_2 \leq W\} \quad (6)$$

なお，実装は $Z_0 = 10$(約 1.3 秒)，$Z_1 = Z_2 = 5$ とし，$Z_3 = 68$(約 4.5 秒) とした．

2.4 グルーピング

類似区間が作る（類似）線分 $\{(t,l,r_s(t,l)) \,|\, t \in [t_1,t_2] \in L(l)\}$ は，l と t の微小な変化にも過敏に反応して，異なる線分を形成している．そこで，t と l の微小変化を許容するように区間統合を考える．

まず，$[t_{1,1},t_{1,2}] \cap [t_{2,1},t_{2,2}] \neq \emptyset$ となる　$[t_{1,1},t_{1,2}] \in L(l)$，$[t_{2,1},t_{2,2}] \in L(l+1)$ に対して，$L(l)$ と $L(l+1)$ の更新処理を次のように行う．区間 $[t_{1,1},t_{1,2}]$ と $[t_{2,1},t_{2,2}]$ を統合した

$$[t_{3,1},t_{3,2}], \text{ s.t. } t_{3,1} = \min\{t_{1,1},t_{2,1}\},\ t_{3,2} = \max\{t_{1,2},t_{2,2}\} \quad (7)$$

を使って，

$$L(l) := L(l) - \{[t_{1,1},t_{1,2}]\},$$
$$L(l+1) := (L(l+1) - \{[t_{2,1},t_{2,2}]\}) \cup \{[t_{3,1},t_{3,2}]\} \quad (8)$$

と更新する．l を 1 から上げていくことでより長い区間に類似線分区間が結合されていく．

次に，l を固定して $L(l)$ の要素である類似区間の統合を考える．任意の $[t_{1,1},t_{1,2}],[t_{2,1},t_{2,2}] \in L(l)$ に対して，これらの区間が次のいずれか一方の条件 1. もしくは 2.

1. $|t_{1,1} - t_{2,1}| \leq \min[\gamma \max\{t_{1,2} - t_{1,1}, t_{2,2} - t_{2,1}\}, Z_4]$
2. $|t_{1,2} - t_{2,2}| \leq \min[\gamma \max\{t_{1,2} - t_{1,1}, t_{2,2} - t_{2,1}\}, Z_4]$

を満たすとき，$[t_{1,1},t_{1,2}]$ と $[t_{2,1},t_{2,2}]$ を (7) 式 で統合し，$L(l)$ を

$$L(l) := (L(l) - \{[t_{1,1},t_{1,2}],[t_{2,1},t_{2,2}]\}) \cup \{[t_{3,1},t_{3,2}]\}$$

と更新する．ここで γ は，$0 < \gamma < 1$ であって，統合を許す区間長の割合を表すパラメータであって，Z_4 は，統合によって区間が膨張し過ぎるのを抑制するパラメータである．更新後の $L(l)$ において，さらに条件 1 または 2 を満たす区間があれば，更新を続け，条件 1 または 2 を満たす区間の組がなくなるまで繰り返す．実装では $\gamma = 0.2$，$Z_4 = 100\gamma = 20$ とした．

最終的に得られた $L(l)$ の要素が同じメロディパターンを構成するとみなす．

2.5 後半の間奏，終奏の削除

$L(l)$ の要素数が同一のメロディパターンの数に相当する．したがって，この要素数が 1 $(\#L(l) = 1)$ のところは，間奏と考えられ，さらに楽曲の後半にのみ現れる繰り返し区間は間奏と終奏からなる区間だとみなして $L(l)$ から削除し，$L(l) = \emptyset$ とする．

2.6 終端の決定

全ての繰り返し区間が再生された直後が圧縮後の楽曲の終端と考えられる．しかし，これは全ての繰り返し区間が正しく検出された場合のみに限られ，メロディ終盤の転調などにより，誤った繰り返し区間が検出されたときは，メロディの途中で楽曲の終端となってしまうことも考えられる．そこで，全ての繰り返し区間が再生された後の次の繰り返し区間が始まる位置，または同時に再生されていた繰り返し区間の終わる位置を終端として定めることにする．

一般性を失うことなく $[t_{1,1}^{(l)},t_{1,2}^{(l)}],\ldots,[t_{p,1}^{(l)},t_{p,2}^{(l)}] \in L(l)$ は，$t_{1,1}^{(l)} < t_{2,1}^{(l)} < \cdots < t_{p,1}^{(l)}$ と仮定してよい．圧縮の際の終端に対応する位置を求めるために，すべてのメロディパターンが一度現れたフレーム t' を調べ，これより先のあるメロディパタンの最初の時刻を終端 E とする．それぞれ，以下の通りである．

$$t' = \max_{\{l \,|\, L(l) \neq \emptyset\}} \{t_{1,2}^{(l)} | [t_{1,1},t_{1,2}] \in L(l)\}, \quad (9)$$

$$E = \min_{\{l \,|\, L(l) \neq \emptyset\}} \{t_{2,1}^{(l)} | t_{2,1} - l > t'\} \quad (10)$$

3 楽曲信号の類似性解析

3.1 一曲ごとの類似性解析

RWC 研究用音楽データベース [5] 115 曲の中からいくつか選び，1 曲ごとに類似性解析した結果を Fig 3 ∼ Fig 7 に示す．第一行（第一レイヤー）は，人間が試聴してメロディの時刻構成を調べた結果であり，同じ濃さの部分

がその人が同じメロディパターンと判断した区間である．第ニから第四レイヤーが提案手法で類似解析した結果である．第ニから第四レイヤーの垂直にまたがる直線が圧縮の終端時刻（E）を示している．

Fig 3「No.1 永遠のレプリカ」はアップテンポのポピュラー音楽であるが，第 5 レイヤーにおけるサビ出だしの区間が 2 番と 4 番でしか検出できていないため，終端までの区間にサビが 2 度再生されてしまう．これは，B メロの終盤からサビの前半という区間構成からなる，最初のサビとの不一致が原因として考えられ，グループ化の際の区間長の更新方法が適さなかったためと考えられる．Fig 7「No.24 It's all right」は電子音が強調されたポピュラー音楽であるが，B メロの繰り返し区間からなる第 7 レイヤーが，1 番の B メロを含んでいないため，終端の位置が 2 番の終わりに近い位置までずれ込んでいる．第 8 レイヤーでは 3 つの B メロ区間が検出されているが，第 7 レイヤーと統合するための許容条件に適さず，それぞれ別のグループと判断されている．グループ化の際，許容条件は区間長の 20%で固定していたが，曲調などの判断基準をもって動的に条件を変えられるアルゴリズムの考案が望まれる．また，全体的に転調への対応が不足していた．サビ区間は比較的検出できているが，転調を伴って歌われる頻度の多い A メロや B メロは内部での繰り返しを検出したり，細かなグループ構成で検出されたりすることが多く，全く検出されていない区間のある楽曲も確認された．

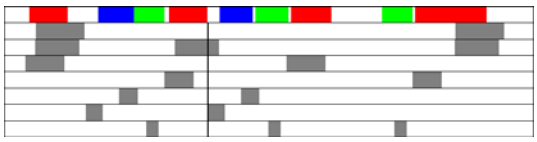

Fig. 3: Result: RWC-MDB-P-2001 No.1 永遠のレプリカ

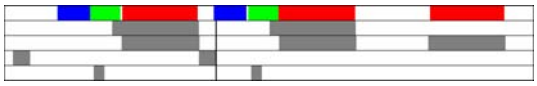

Fig. 4: Result: RWC-MDB-P-2001 No.5 恋の Ver.2.4

3.2　類似性解析によるデータベース全体の圧縮

RWC 研究用音楽データベース [5] から，ポピュラー音楽 100 曲 (RWC-MDB-P-2001　No.1～No.100) と，著作権切れ音楽 15 曲 (RWC-MDB-R-2001　No.1～No.15) の

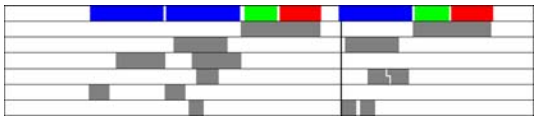

Fig. 5: Result: RWC-MDB-P-2001 No.10 Getting Over

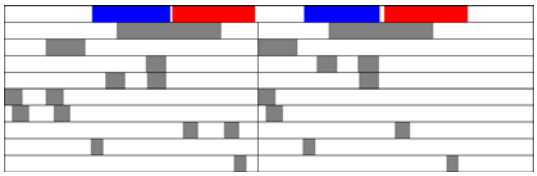

Fig. 6: Result: RWC-MDB-P-2001 No.12 KAGE-ROU

合計 115 曲を用いた圧縮実験を行った．本実験でも圧縮のための類似性解析は一曲ごとに行っている．

データベース全体に対する圧縮率は 46.2 % であり，一曲あたりの平均は 45.7 %，標準偏差は 12.6 % であった．基本統計量と分布を Table 1 と Fig.8 に示す．ここでの圧縮率は (圧縮後フレーム)/(圧縮前フレーム) である．全体で 46.2%に圧縮されているため，検索の際は約 2 倍の高速化が実現できる．

4　まとめと今後の課題

本稿ではクロマベクトルを用いて楽曲の繰り返し区間を検出し，楽曲信号の圧縮手法を提案した．RWC 研究用音楽データベース 115 曲を用いて本手法を評価した結果，データベースを約 45%まで縮小することで，検索の 2 倍の高速化を実現した．本手法の発展として，工藤 [4] では，今回の類似区間の情報を利用して楽曲データベースの特徴量を再構成する方法の提案も行い，s-CDP での検索実験についても報告している．

今後の課題は，繰り返し区間の検出時における楽曲の転調への対応，グルーピングの際の類似度を考慮した区間長の設定，終端決定の新たな手法の考案などが挙げられる．なお，転調への対応法としては，RefraiD[6] でクロマベクトルの要素をシフトされながら類似度を計算する方法が提案されている．これらを踏まえたて，人間が同じと考えた類似区間との適合の評価も行う必要がある．

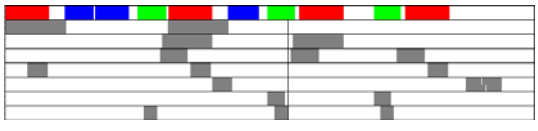

Fig. 7: Result: RWC-MDB-P-2001 No.24 It's all right

Table 7: Compression of RWC music database

圧縮前総フレーム	411980(約 7 時間 20 分)
圧縮後総フレーム	186556(約 3 時間 20 分)
圧縮率	46.2%
(一曲あたり)	
平均	46.7%
標準偏差	12.6%
最小	25.6%
最大	100%

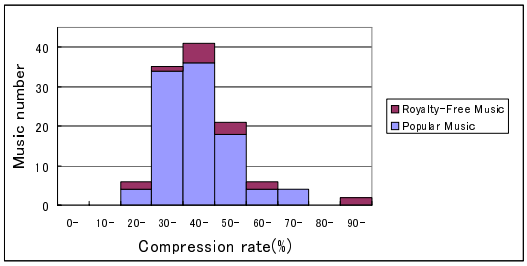

Fig. 8: Histogram of compression ratios

謝辞

本研究の一部は，平成 16 年度埼玉大学 21 世紀総合研究プロジェクト経費による支援を受けて行われた.

参考文献

(1) Barsch, M.A. and Wakefield, G.H., To catch a chorus: Using chroma-based representations for audio thumbnailing, Proc. WASPAA '01, pp.15-18, 2001.

(2) Cooper, M. and Foote, J., Automatic music summarization via similarity analysis, Proc. ISMIR 2002, pp.81-85, 2002.

(3) Dannenberg, R.B. and Hu N., Pattern discovery techniques for music audio, Proc. ISMIR 2002, pp.63-70, 2002.

(4) 工藤雅志, クロマベクトルを用いた楽曲信号の圧縮と s-CDP 検索への応用, 2004 年度埼玉大学工学部情報システム工学科卒業論文, 2005.

(5) 後藤真孝, 橋口博樹, 西村拓一, 岡　隆一, RWC 研究用音楽データベース：ポピュラー音楽データベースと著作権切れ音楽データベース, 情処研報音楽情報科学, 2001-MUS-42-6, pp.35-42, 2001.

(6) 後藤真孝, リアルタイム音楽情景記述システム:サビ区間検出法, 情報処理学会　音楽情報科学研究会　研究報告, 2002-MUS-47-6, Vol.2002, No.100, pp.27-34, 2002.

(7) 後藤真孝, SmartMusicKIOSK：サビ出し機能付き音楽視聴機, 情報処理学会　インタラクション 2003 論文集, pp.9-16, 2003.

(8) 後藤真孝, 平田圭二, 音楽情報処理の最近の研究, 日本音響学会誌, Vol.60, No.11, pp.675-681, 2004.

(9) Nishimura,T., Hashiguchi, H., Takita, J., Zhang, J. X, Goto, M. and Oka, R., Music Signal Spotting Retrieval by a Humming Query Using Start Frame Feature Dependent Continuous Dynamic Programming, Proc. ISMIR 2001, pp.211–218, 2001.

(10) 西村拓一, 橋口博樹, 関本信博, 張建新, 後藤真孝, 岡隆一, 始端特徴依存連続 DP を用いた鼻歌入力による楽曲信号のスポッティング検索の高速化, 情報処理学会音楽情報科学, Vol.42-2, pp.7-14, 2001.

(11) 橋口博樹, 西村拓一, 張　建新, 滝田順子, 岡隆一, モデル依存傾斜制限型の連続 DP を用いた鼻歌入力による楽曲信号のスポッティング検索, 電子情報通信学会論文誌, Vol.J84-D-II, No.12, pp.2479-2488, 2001.

(12) 大津展之, 判別および最小 2 乗法基準に基づく自動しきい値選定法, 信学論 (D), J63-D, 4, pp.349-356, 1980.

(13) Peeters, G., Burthe,A.L. and Rodet, X., Toward automatic music audio summary generation from signal analysis, Proc. ISMIR 2002, pp.94-100, 2002.