人間の感性のスコアリングに基づくマルチメディア情報検索の基礎研究

A Study on Multimedia Information Retrieval based on Human Scoring

橋口博樹　情報システム工学科　講師
Hiroki Hashiguchi, Department of Information and Computer Sciences

1　はじめに

本研究では，マルチメディア情報検索分野において，ユーザが似ていると感じるマルチメディア情報の類似性と，情報検索システムでモデル化されている類似性がどの程度合致しているか否かを定量的な実験を通して検証するためのシステム開発である．この検証結果を検索システムにフィードバックすることで，より人間の感性に近い検索結果をもたらすマルチメディア検索システムを構築することを目的としている．今年度はマルチメディアとして画像と音楽音響信号を題材とした．まず, 画像の類似性解析では，コンピュータが提示した複数のクエリ画像に対してユーザが所望する画像との類似度をユーザ自身が与え，それらをもとに画像検索を行うアプリケーションの開発に着手し，インターフェース部を完成させた．さらに，類似性解析によって得られる特徴量と人間の感性とどれほど似ているかを検証するために，回帰分析を応用した検索手法の実験を行った．インターフェース部には絞り込み検索を容易に行うため，一度検索した画像をクエリ画像として取り入れる機能を実装している．１００枚程度の画像を用いた小規模な実験を通して、数回のブラウジングで所望の画像まで絞りこめることを確認した．

一方，ポピュラー音楽の楽曲信号を用いた研究では，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とを比較した．これは, メロディが類似している区間を統合し，楽曲を自動縮約することにも応用できる．類似性を解析するための特徴量には既存の研究でポピュラー音楽の場合有効とされているクロマベクトルを用いた．実際に RWC 研究用音楽データベース 115 曲を用いて縮約データベースを構築し約半分に圧縮できることを確認した．さらに楽曲検索も行い，検索時間が約半分に短縮されることを確認した．検索精度については十分な実験を行っていないが，数例については縮約しない場合と比較してさほど低下しなかったことを確認している．2 節においてこの楽曲信号の解析について報告する．

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，区間ごとの特徴パターンを統合するとともに，間奏や後奏を除いた繰り返し区間のみで楽曲を構成する．楽曲信号からなる特徴パターンに対して「前奏 (イントロ) → A メロ → B メロ → サビ」のように再構成を行うことを楽曲信号の圧縮と定義する．

2.2　クロマ類似度の計算

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

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

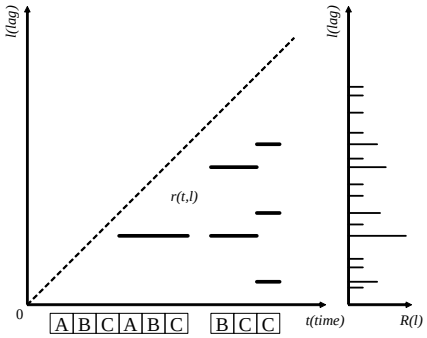
と定義し，$\vec{v}(t) = (v_1(t), ..., v_{12}(t))$ をクロマベクトルと呼ぶ [3].

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，

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

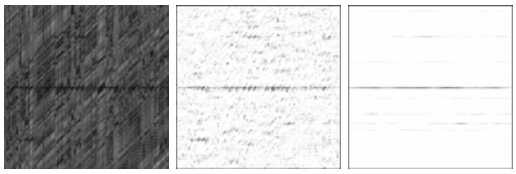
と定義する [3]．ここで，ベクトルのノルム $|\cdot|$ は通常のユークリッドノルムとする．分母の $\sqrt{12}$ は，1 辺の長さが 1 の 12 次元超立方体の対角線の長さに対応し，$r(t,l)$ を $0 \leq r(t,l) \leq 1$ に正規化している．

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) が図 1 のように右下半分の三角形領域に現れる．図 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 [3] 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 [3] の方法でノイズ処理し，それを $r_d(t,l)$ と書く．図 2 の中央には左図のノイズ処理後の $r_d(t,l)$ を示す．



A:melodyA　B:melodyB　C:Chorus

図 1: クロマ類似度 $r(t,l)$ の概念図



(左図) $r(l,t)$ の一部
(中央) 左図のノイズ処理：$r_d(t,l)$ の一部
(右図) $L(l)$ に残った線分（類似線分抽出）

図 2: 線分の強調処理

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

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

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

図 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．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の区間 $(2Z+1$ 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 (図 1 右部分).

$$R(l) = \sup_{l \leq t \leq W} \int_{t-Z_0}^{t+Z_0} \frac{r_d(\tau, l)}{2Z_0} d\tau \approx \max_{l \leq t \leq W} \frac{1}{2Z_0+1} \sum_{\tau=\max\{0, t-Z_0\}}^{\min\{t+Z_0, W\}} r_d(\tau, l) \tag{3}$$

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．

$$R'(l) = \max\left\{0, R(l) - \int_{l-Z_1}^{l+Z_1} \frac{R'(\xi)}{2Z_1} d\xi\right\} \approx \max\left\{0, R(l) - \frac{1}{2Z_1+1} \sum_{\xi=\max\{0, l-Z_1\}}^{\min\{l+Z_1, W\}} R(\xi)\right\} \tag{4}$$

$\{R'(1),\ldots,R'(W)\}$ 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 [8] 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，$r_s(t,l)=0$ とすることで線分の強調を行う．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 (5) 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．

$$L(l)=\{[t_1,t_2]\,|\,r_s(t,l)>\beta\text{ for }\forall t\in[t_1,t_2],t_2-t_1>Z_3,0\leq t_1<t_2\leq W\}\tag{5}$$

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

$$[t_{3,1},t_{3,2}],\ \text{s.t.}\ t_{3,1}=\min\{t_{1,1},t_{2,1}\},\ t_{3,2}=\max\{t_{1,2},t_{2,2}\}\tag{6}$$

を使って，

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

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 (6) 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．

2.4.1 後半の間奏，終奏の除去

$L(l)$ 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．

2.4.2 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．

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

2.5 圧縮結果

RWC 研究用音楽データベース [2] から,「RWC-MDB-P-2001 No.5 恋の Ver.2.4」の圧縮結果を図 3 に示す．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，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．

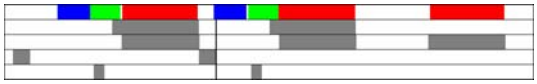


図 3: 圧縮結果:RWC-MDB-P-2001 No.5 恋の Ver.2.4

さらに RWC 研究用音楽データベース [2] 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) の合計 115 曲を用いた実験を行った．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，46.7 % 標準偏差は 12.6 % であった．ここでの圧縮率は (圧縮後フレーム)/(圧縮前フレーム) と定義する．全体で 46.2%に圧縮されているため，検索の際は 2 倍以上の高速化ができる．

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] 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.

[2] 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.

[3] 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.

[4] 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.

[5] 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.

[6] 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.

[7] 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.

[8] 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.