音楽情報科学 30－6
（1999．5．29）

時間正規化を用いたハミング検索システム

西原 祐一　小杉 尚子　紺谷 精一　山室 雅司

NTTサイバースペース研究所
〒239-0847 横須賀市光の丘 1-1
E-mail: nishihar@isl.ntt.co.jp

概要： ハミングを用いた音楽検索システムにおいて，従来は音符を単位としたマッチング方式が主に使われてきた．それに対し，われわれは時間を単位としたマッチング方式を開発した．この方式は，マッチングをする際にハミングおよびデータベース中のオリジナル曲の，時間の単位を正規化した上でマッチングを行なうものである．この方式は，音符抽出誤りに対する耐性が高く，また，データベースのインデックス化が可能なために，高速な検索が可能である．この方式を実装した実験システムを作成し，実験の結果について報告する．

Humming Query System Using Normalized Time Scale

Yuichi Nishihara, Naoko Kosugi, Seiichi Kon'ya, Masashi Yamamuro

NTT Cyberspace Laboratories
1-1 Hikari-no-oka, Yokosuka-shi, Kanagawa 239-0847, JAPAN
E-mail: nishihar@isl.ntt.co.jp

Abstract: The traditional approach to music retrieval systems using humming as their query is 'note-based': series of notes are extracted from the humming and either DP matching or its alternatives are performed. However, we have developed a new 'time-based' approach whose basic concept is to normalize the time scale of the humming and the music in the database. The new method has high robustness against note extraction errors, and also high retrieval speed is achieved since indexing of the database is possible. We have built an experimental system applying the new method and the result of the preliminary experiment is shown.

1　はじめに

近年，マルチメディアデータベースに対する内容検索の要望が高まっており([1])，音楽データベースに対しても，ハミングを検索キーとするハミング検索の研究がいくつか行われている([2-8]).

図 1 にハミング検索システムの基本構造を示す．ハミング検索システムにおいては，データベース中の曲およびハミングから数々の特徴量を抽出し，その特徴量を用いて類似検索を行なう．類似検索を行なうのは，以下の理由により，たとえ同じ曲であってもデータベース中のオリジナル曲とハミングから抽出した特徴量が必ずしも一致しないからである．

- 利用者が，オリジナル曲の通りに，曲を記憶しているとは限らない
- 利用者が，オリジナル曲の通りに，ハミングできるとは限らない
- 特徴量を抽出する際に，誤差が生じる可能性がある

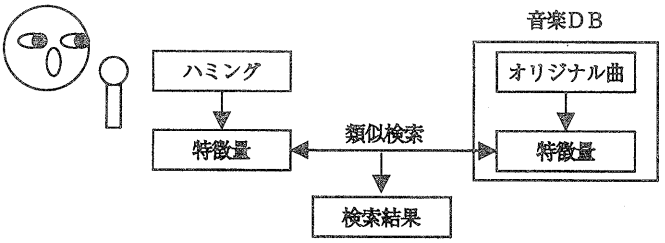

図 1 ハミング検索システムの基本構造

この，オリジナル曲と，ハミングとの不一致は，以下のように分類することができる．

- 音程のずれ：絶対音程のずれ（＝調のずれ）／相対音程のずれ（＝音程の狂い）
- 音長のずれ：絶対音長のずれ（＝テンポのずれ）／相対音長のずれ（＝テンポのゆらぎ）
- 音符数のずれ：音符の挿入／音符の削除
- 不特定位置からのハミング開始

これらのずれをある程度許容することによって，類似検索は実現される．なお，オリジナル曲とハミングとの不一致については，文献[7,8]に詳述した．

従来のハミング検索の研究では，音符を単位としたマッチングが行われてきた．図2(a)に，音符を単位としたマッチングの例を示す．一般にハミングとオリジナル曲のテンポはずれており，このような場合，音符を単位としたマッチングは有効である．しかし，図のように，音符の挿入・削除がある場合，比較する音符同士の対応関係がずれてしまうために正確なマッチングができなくなる．

そこで，音符数のずれを許容可能とするマッチング手法として，DP マッチングが主に用いられてきた．しかし，DP マッチングでは，データベースのインデックス作成ができず，データベース中の音符数 n に対し，計算時間が $O(n)$以上かかるため，大規模データベースには不向きである．

音符を単位としたマッチングで DP マッチングに代わるものもいくつか提案されている([6,7])．これらは，いずれも DP マッチングより高速な検索が可能であるが，DP マッチングより精度が低いか，許容できる音符数のずれに制限がある．

そこで，われわれは，音符数のずれに対する耐性の強いマッチング方法として，時間を単位としたマッチングを提案する．

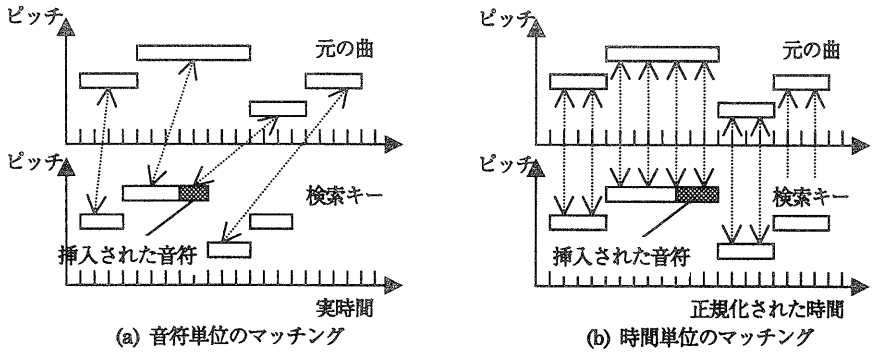

図 2 音符単位のマッチングと時間単位のマッチング

2　時間を単位としたマッチング

図2(b)に，時間を単位としたマッチングの例を示す．

ハミングと，オリジナル曲のテンポが同じであれば，時間を単位とした，時系列データの比較が可能となる．例えば，二つの時系列データ A と K を比較する場合，まず，比較を行なうための時間の単位 Δt を設定し，時間 $\Delta t, 2\Delta t, \ldots, n\Delta t$ におけるピッチを測定し，$A(a_1, a_2, \ldots, a_n)$，$K(k_1, k_2, \ldots, k_n)$ のようにベクトル値として表わす．そして，時系列データ A，K の類似度は，例えばこの二つのベクトルのユークリッド距離 $\|\boldsymbol{A}-\boldsymbol{K}\|$ を計算することにより求めることができる．

テンポが同じ曲同士のマッチングであれば，「絶対音長のずれ」は存在しない．また，「音符数のずれ」は，すべて「音程のずれ」もしくは「相対音長のずれ」に帰着できる．

この時間を単位とするマッチングは，「音符数のずれ」の影響を受けないために，音符を単位とするマッチングと比べて検索精度が高くなることが期待される．また，特徴量を，多次元ベクトル空間上の点として表現できるために，データベースのインデックス作成も可能となり，検索速度も速くなることが期待される．

ただし，本方式を用いるためには，オリジナル曲とハミングのテンポをそろえる必要がある（われわれは，これを「時間正規化」と呼ぶことにする）．また，「絶対音長のずれ」および「音符数のずれ」以外のずれを解消する必要もある．

図3に，時間を単位とするマッチングを用いたハミング検索システムの処理の流れを示す．この個々の行程について，以下の小節において解説する．なお，処理の流れの詳細については，文献[8]参照．

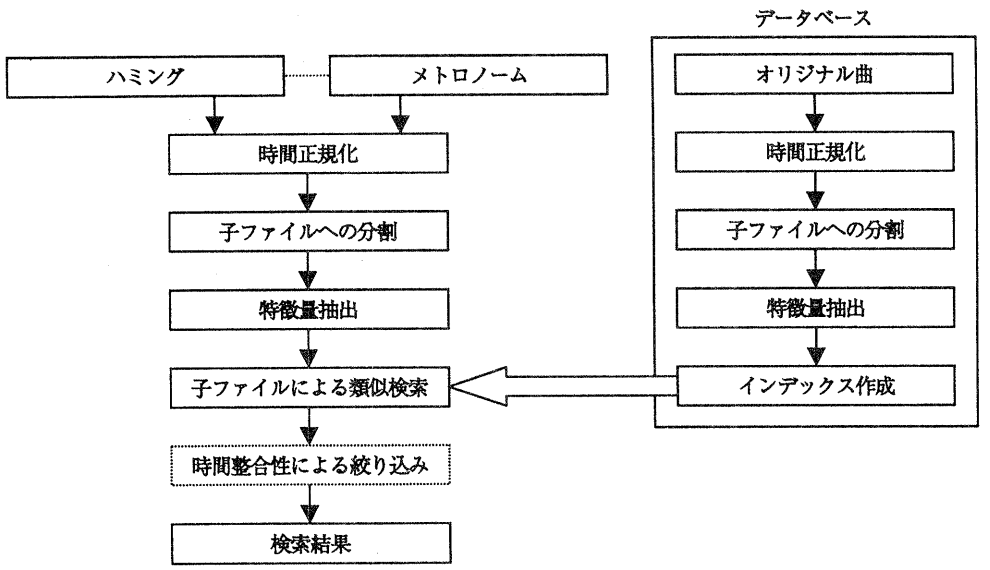

図 3 時間を単位としたマッチングの処理の流れ

2.1　時間正規化

2.1.1　　データベースの時間正規化

現在，データベース中のオリジナル曲としては，MIDI データを利用している。MIDI データを利用する限り，データベースの時間正規化は容易である．MIDI データにおいては，音符の長さは拍子に対する相対的な長さで記述されており，実際の演奏速度は，ヘッダに記述されている，一分間に刻まれる

拍子の数（＝テンポ）によって決まる．したがって，MIDI データにおいては，拍子を基準にすれば時間はあらかじめ正規化されていると考えてよい．

2.1.2　ハミングの時間正規化

ハミングの時間正規化を行なうためには，ハミングからテンポを抽出する必要がある．これをハミングのデータから直接行なうことは以下の理由から難しい．

・ 利用者が一定のテンポでハミングできるとは限らない

・ ハミングが短すぎて，テンポを抽出するためには情報が不十分な可能性がある

しかし，ユーザインタフェースに，メトロノームを付け加え，利用者にメトロノームに合わせてハミングしてもらうことにより，テンポの抽出は容易に可能となる．このメトロノームは，定められたテンポに従い拍子音を鳴らしたり，光を点滅させたりするものなど，利用者がそれに合わせてハミングできるようなものであれば何であってもよい．また，利用者が，ハミングしたい曲に合わせて自由にテンポを調整することができるようにすることもできる．また，利用者に，ハミングに合わせてマウスのクリックやキーボードの押下によって拍子を刻んでもらい，その信号をハミングとともに記録することによって，ハミングのテンポを知るようにすることも可能である．

上記のような手法により，ハミングの時間正規化を行なう．

2.2　部分曲を使ったマッチング

曲のどの部分がハミングされたかを高速に特定するために，部分曲を使ったマッチングを行なう．これには，まず，時間正規化された後のオリジナル曲とハミングを，スライディング・ウィンドウ方式によって部分曲に分割する．部分曲の長さおよび，隣接する部分曲同士の間隔（分割ずらし幅）はパラメータとなる．データベースの部分曲については，高速な検索が可能となるうようインデックスを作成，ハミングの部分曲との間で類似検索を行なう．この類似検索の結果，選出されるデータベースの部分曲の間で時間整合性による絞り込みを行い，最終的に，ハミングにマッチするオリジナル曲の部分を特定する．

2.3　ヒストグラムの利用

部分曲同士のマッチングを行なう際，特徴量として第 2 節で解説したベクトル化された時系列データだけでなく，ヒストグラムも利用する．このヒストグラムとは，部分曲中に出現する各ピッチの時間的頻度を記録したものである．このヒストグラムをベクトルで表わし，類似度をベクトル間の距離で表わすことにより，「相対音長のずれ」を吸収できるようになる．

また，「絶対音程のずれ」を吸収するために，平均値による正規化を行なったヒストグラム，部分曲の最初の音を基準としたヒストグラム，音程の推移のヒストグラムも作成する．また，「相対音程のずれ」を吸収するために，分布を平滑化させたヒストグラムも作成する．

2.4　インデックスの作成および類似検索

上記ベクトル化された特徴量に対し，高速検索を可能とするためのインデックスを作成する．インデックスは，ExSight 画像検索システム([10])と同様，VAMSplit R-tree([11])およびその改良版である C-tree([12])を用いる．また，類似検索を行なう DBMS としては，HyperMatch エンジン([12])を利用する．

2.5　時間整合性による絞り込み

時間整合性の概念については，文献[7,8]に記した．ハミングの部分曲 K_1 に対して，オリジナル曲 A の部分曲 A_p がヒットしたとする．さらに，ハミングの部分曲 K_2 に対して，以下の条件を満たす部分曲 X がヒットした場合に，部分曲 A_p の K_1 に対する類似度を高くする．

・ Xは，オリジナル曲Aの部分曲（A_qとする）である
・ K_1, K_2, A_p, A_qの開始時間が，正規化された時間で$t_{K1}, t_{K2}, t_{Ap}, t_{Aq}$と表わされたとすると，$t_{Aq}-t_{Ap}=t_{K2}-t_{K1}$である

3 実験システム

時間を単位としたマッチング方式の有効性を確かめるための実験システムを実装した．ただし，現時点では第 2 節に記した特徴の一部しか実装されていない．以下，現在の実験システムの構成および評価実験の結果について記す．

3.1 実験システムの構成

実験システムの特徴は以下の通りである．

・ システムは，クライアント（PentiumII 300MHz, Windows98）およびサーバ（Ultra SparcII 300MHz x 2, Solaris2.6）から成る
・ データベースに含まれる曲は 1200 曲であり，市販の MIDI データから旋律チャネルのみを抜き出して用いた．旋律チャネルの一曲あたりの平均音符数は 357 であった
・ ハミングからの音程抽出には，市販の作曲用ソフトを用いた
・ 時間整合性についての実装はまだ行われていない．ハミングの部分曲に対してヒットする，複数のデータベース中の部分曲については，単純な OR 演算を施しているのみである
・ 部分曲作成のためのスライディング・ウィンドウのパラメータは，部分曲の長さを 20 拍，ずらし幅を 4 拍とした

3.2 実験の方法および結果

実験は，11 人（男性 10 人，女性 1 人）の被験者に，各々3 曲以上の曲を，データベースに含まれる 1200 曲の曲名一覧の中から任意の曲を選んでもらい，曲の任意の部分を記憶から歌ってもらって行なった．また，検索を行なう前に，ハミングからの音程抽出の結果を聞いてもらい，抽出の結果に満足できるかどうかを 5 段階（5＝良い，4=やや良い，3=普通，2=やや悪い，1＝悪い）で評価してもらった．

全結果			音程抽出の評価が「普通」以上の結果		
検索数	1位正答数	3位以内正答数	検索数	1位正答数	3位以内正答数
53	42 (79%)	43 (81%)	32	29 (91%)	29 (91%)

図 4 実験結果

図 4 に，実験の結果を記す．図中，1 位正答数，3 位以内正答数とは，被験者が意図した曲が検索結果の各々1 位，3 位以内に現れた回数を表す．音程抽出の良し悪しに関わらず検索を行った場合でも，正答率は 80%近く，また，音程抽出の評価が「普通」以上の場合，正答率は 90%を超えた．

検索にかかる時間については，市販の作曲用ソフトによるハミングからの音程抽出はリアルタイムで行われ，検索そのものには，通信にかかる時間を除き，平均で 3 秒程度かかっている．

4 まとめ

本稿において，われわれはハミング検索システムにおいて，音符単位のマッチングに代わる手法として時間単位のマッチングを提案した．この方式は，時間正規化を行うことによりテンポの違いを吸収するものであるが，音符単位のマッチングと比較して以下の点が優れている．

・「音符数のずれ」の影響を受けないため，高い検索精度が期待できる
・データベースに対してインデックス作成が可能なため，高速な検索が可能である

そして，提案したシステムの有効性を試すための実験システムを構築し，評価実験を行った．同じ実験システム上で音符単位のマッチングを実装していないために，比較評価は得られていないが，単独評価としては良好な結果が得られており，今後は本方式のより完全な実装を行っていく予定である．

今回の実装においては，ハミングそのものからテンポ抽出を行うことができなかったために，メトロノームという補助手段を用いた．メトロノームに合わせてハミングするというユーザインタフェースの良し悪しについては，今後主観評価などの評価を行っていく必要がある．一方で，音符単位のマッチングでも，音符抽出誤りを減らすために，音符の始まりと終わりが認識しやすくなるようなハミングのしかたをする必要があるなど，特殊な歌い方を利用者に要求している．

今後の課題としては，以下が挙げられる．

・時間整合性の実装
・部分曲に分割する際のパラメータ，およびその他のパラメータの見なおし
・ピッチ抽出ルーチンの自作．特別な方法でハミングせずともピッチ抽出が可能かどうか見極める

参考文献

[1] 串間和彦，赤間浩樹，紺谷精一，山室雅司：「色や形状等の表層的特徴量にもとづく画像内容検索技術」，情報処理学会論文誌，Vol.40 No.SIG3(TOD1) pp.171-184, 1999

[2] 園田智也，後藤真孝，村岡洋一：「WWW上での歌声による曲検索システム」，電子情報通信学会技術報告，SP97-103，1998

[3] 貝塚智憲，後藤真孝，村岡洋一：「歌声の旋律情報と歌詞情報をキーとした曲検索システム」，情報処理学会第54回全国大会，7J-6，1997

[4] 蔭山哲也，高島羊典：「ハミング歌唱を手掛りとするメロディ検索」，電子情報通信学会論文誌，Vol.J77-D-II No.8 pp.1543-1551，1994

[5] A.Ghias, J.Logan, D.Chamberlin, and B.C.Smith: *Query By Humming – Musical Information Retrieval in An Audio Database*, ACM Multimedia 95, Electronic Proc., 1995

[6] R.J.McNab, L.A.Smith, D.Bainbridge, and I.H.Witten: The New Zealand Digital Library MELody inDEX, D-Lib Magazine (http://www.dlib.org/dlib/may97/meldex/05witten.html), 1997

[7] 西原祐一，梅田昌義，紺谷精一，山室雅司，福本誠：「大規模音楽ＤＢに対する高速ハミング検索方式」，アドバンスド・データベース・シンポジウム '98，pp.117-124, 1998

[8] Y.Nishihara, N.Kosugi, M.Umeda, S.Kon'ya, and M.Yamamuro: *Humming Query System Using Normalized Time Scale*, Proc. of CODAS'99 (to be published), 1999

[9] J.Foote: *An overview of audio information retrieval*, Multimedia Systems, Vol.7, pp.2-10, 1999

[10] 串間和彦，赤間浩樹，紺谷精一，木本晴夫，山室雅司：「オブジェクトに基づく高速画像検索システム：ExSight」，情報処理学会論文誌，Vol.40 No.2 pp.732-741, 1999

[11] D.A.White and R.Jain: *Similarity indexing: algorithms and performance*, Proc. of the SPIE: Storage and Retrieval for Image and Video Database IV, pp.62-75

[12] K.Curtis et al.: A Comprehensive Image Similarity Retrieval System that Utilizes Multiple Feature Vectors in High Dimensional Space, Intl. Conf. on Information, Communications and Signal Processing, ICICS '97, pp.180-184