Earth Mover's Distanceを用いたハミングによる類似音楽検索手法

獅々堀 正幹[†] 大西 泰代[††]
柘植 覚[†] 北 研二[†††]

近年，楽曲配信サービスの普及により，容易に音楽データをダウンロードして試聴できるようになった．しかし，サーバ側で蓄積している音楽データが膨大になるにつれ，音楽データに対する効率的な検索手法が必要になっている．特にハミングを入力とする検索手法が近年活発に研究されており，音楽特徴量間の類似度計算に DP マッチングやユークリッド距離を用いる手法が主流であった．本論文では，距離尺度として Earth Mover's Distance（EMD）を用いたハミング検索手法を提案する．EMD は輸送問題における輸送コストの最適解であり，本手法では輸送問題における各供給地が有する資源量を各音符の音長，輸送コストを各音符の出現時間と音高情報から算出することで，リズムと音程との類似度を同じ距離尺度で計り，全体の曲調が類似した曲を検索する．さらに，EMD の計算量が音符数に対して指数関数的に増加することに着目し，検索精度を維持しつつ計算コストを低減可能な音楽特徴量を提案する．約 500 曲の音楽データベースに対してハミングデータ 40 曲を入力とした評価実験を行った結果，ユークリッド距離を用いる手法より検索結果上位 10 位以内に正解データが出現する割合が約 30%向上した．また，DP マッチングを用いる手法と比べて，極端に音高の外れた音符を含むハミングデータに対する柔軟性を確認した．

Similar Music Retrieval for the Query-by-humming Using the Earth Mover's Distance

Masami Shishibori,[†] Yasuyo Ohnishi,[††] Satoru Tsuge[†] and Kenji Kita[†††]

Music retrieval systems are extremely useful for collecting digital music data from on-line music distribution sites. Especially, there is a great need to develop effective techniques for content-based music retrieval systems, which can retrieve by humming query. The main issues in this research is how to decide the similarity of each music features extracted from music data. In order to calculate the similarity, some conventional methods use Euclid distance or DP matching, but it is very hard to solve the problem of the vagueness of humming query. In this paper, we propose a new similar music retrieval method based on humming query using the Earth Mover's Distance as the distance measure. Computing the EMD is based on a solution to the transportation problem, and the EMD is applied as the distance measure on similar image retrieval systems. In addition, we focus that the time complexity of the EMD is exponential worst case toward the number of notes, the improved method to decrease the number of notes in the music feature is also proposed. Experimental results show that the proposed method can improve the retrieval precision of conventional systems.

1. はじめに

近年，インターネットの普及により，楽曲配信サービスを行うサイトも多数出現し，ユーザは容易に音楽データをダウンロードして試聴できるようになった．しかし，サーバ側で蓄積される音楽データが膨大になるにつれ，ユーザは目的の曲を見つけ出すことが困難となり，音楽データに対する効率の良い検索手法が必要になっている[1)]．

音楽データに対する検索方式は，曲名，歌手名や歌詞等を入力とするキーワード型の検索方式が一般的であるが，より汎用的なコンテンツ型の検索方式が注目されている[2)]．コンテンツ型検索方式は，ユーザが曲

† 徳島大学大学院ソシオテクノサイエンス研究部
Institute of Technology and Science, Tokushima University
†† 徳島大学大学院工学研究科
Graduate School of Engineering, Tokushima University
††† 徳島大学高度情報化基盤センター
Center for Advanced Information Technology, Tokushima University

の一部のリズムを入力するシステム[3),4)]，実際に歌ってハミングを入力するシステム[5)~17)]，感性情報により検索するシステム[18)]，ユーザの嗜好に基づいて検索するシステム[19)] 等が存在する．特にハミング検索システムは，曲名や歌手名を記憶していなくても，メロディだけを覚えていれば検索できるため，発展が期待されている[2)]．このような背景から，我々はコンテンツ型検索方式の中でも，特にハミングを入力としてMIDI 形式の音楽データを検索する研究を行っている．

従来のハミング検索システムとしては，量子化した音楽データに対して n-gram 検索を行う手法[5)]，ハミングデータ内に発生する誤りパターンを HMM で学習し，確率的に類似性を判定する手法[6),7)]，音楽データの音長や音高の変化の類似性を DP マッチングを用いて判定する手法[5),8)~12),14),20)]，音高の変化を固定長のベクトルで表現し，ベクトル間の類似性をユークリッド距離で判定する手法[15),16)] 等が存在する．このように様々な距離尺度を用いた手法が提案されているが，音長と音高等の様々な音楽特徴量の類似性を同じ範疇の距離尺度で判定することが重要である．文献 12)，20) では，音長と音高の距離の総和を用いて DP マッチングを行うことで，単独の特徴量を用いた場合よりも大幅な精度向上に成功している．

本論文では，距離尺度として Earth Mover's Distance（EMD）[21)] を用いたハミングによる類似音楽検索手法を提案する．EMD は線形計画問題の 1 つである輸送問題における輸送コストの最適解を求めるアルゴリズムである．また，Rubner ら[21)] により類似画像検索の距離尺度に適用され，検索精度の向上に成功している．EMD を用いた類似画像検索では，画像を色情報により領域分割し，各領域を供給地，各領域の広さを供給地が有する資源量と見なし，各領域の重心点，平均色等の特徴量から輸送コストを定義することで，画像全体の構図が類似した（同じ位置に類似した色が配色された）画像を検索できるといった長所を持つ．

本手法では，音楽データを固定長のメロディ片に分割した後，メロディ片内の各音符を供給地，各音符の音長を供給地の資源量と見なし，各音符の出現時間，音高等の特徴量から輸送コストを算出する．この音楽特徴量を EMD で距離計算することで，リズムと音程との類似度を同じ距離尺度で計り，全体の曲調が類似したメロディを検索する．一方，供給地数を N とすると，EMD の計算量は N に対して指数関数的に増加する．すなわち，EMD を音楽特徴量に適用した際には，計算量が音符数に対して指数関数的に増加する．そこで本手法では，音符数を削減した音楽特徴量に改良することで，検索精度を維持しつつ計算量を低減する．

以下，2 章で既存のハミング検索手法の概要と問題点について述べ，3 章で EMD を用いたハミング検索手法を提案し，4 章で高速化への改善手法を提案する．5 章において実際のハミングデータを用いた実験結果を示し，考察を述べる．最後に 6 章において，まとめおよび今後の課題について述べる．

2. ハミングによる類似音楽検索手法

ハミングによる類似音楽検索システムでは，**図 1** に示すように，ユーザは検索したい曲の一部をハミングする．次に，ハミングデータ自体，もしくはハミングデータを量子化した値から音長や音高といった音楽特徴量を作成する．文献 12)，14) では，入力ハミングデータを単純に MIDI データに変換せず，MIDI データよりも細かい精度で量子化した数値から音楽特徴量を作成している．これに対して本論文では，距離尺度の効果に焦点を絞るため，入力ハミングデータを独自の方法で量子化せず，採譜ソフトで変換した MIDI 音楽データを入力とした．検索対象の音楽データに対しては，あらかじめ抽出した主旋律に対する音楽特徴量データベースを作成し，入力ハミングの音楽特徴量との距離計算を行い，距離の近い類似したメロディを持つ曲を検索する．検索キーがハミングになるため，音程のずれやリズムの違いの吸収等が重要な課題になる[1)]．

2.1 従来のハミング検索手法の分類

従来のハミング検索手法としては，ハミングデータを量子化して n-gram 検索を行う手法[5)]，HMM を用いた学習ベースの手法[6),7)]，DP マッチングを用いる手法[5),8)~12),14),20)]，ユークリッド距離を用いる手

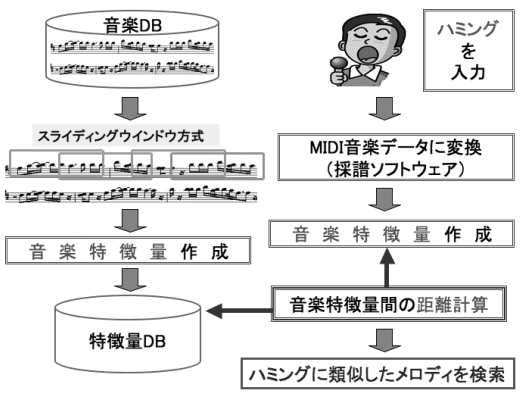


図 1 ハミングによる類似音楽検索システム
Fig. 1 Similar music retrieval systems based on humming query.

法[15),16)] に大別できる．

なかでも DP マッチングを用いる手法は，学習データを用いずに高い検索精度を得ることができるため，数多くの研究成果が報告されている．文献 17) では，DP マッチング，DTW，HMM の 3 種類の検索手法を比較した結果，実用性の面で DP マッチングが最も適していると報告している．また，一般に音高差の音楽特徴量を量子化した系列に対して DP マッチングを行う手法[8)] が多いが，より検索精度を高めるために，音長や音高等の複数の音楽特徴量を統合して DP マッチング[12),20)] を行う手法も提案されている．さらに，音符の削除や挿入に対してペナルティを付加することが可能であり，学習データから適切なペナルティ値を求め，精度向上を図ることもできる[9)]．

ただし，入力系列長 M，N に対する DP マッチングの計算量が $O(MN)$ となるため，大規模なデータに不向きといった欠点がある．DP マッチングの検索コストを削減する研究として，量子化した音楽特徴量を入力として大まかな検索を行った後，絞り込まれたデータに対してのみ DP マッチングを適用する 2 段階処理で高速化を図る手法[5),10)]，また，データベース内の曲を細かいメロディ片に分割し，あらかじめメロディ片単位で DP マッチングを行うことで索引を形成する手法[11)] 等が考案されている．

一方，ユークリッド距離を用いる手法[15),16)] は，音高情報を音高推移特徴ベクトルと呼ばれる固定長ベクトルで表現し，ベクトル間の類似性をユークリッド距離で計算する．ユークリッド距離を用いることで，ユークリッド空間上のデータに対する索引化技術[22)] が適用できる．大規模なデータに対する検索速度の耐久性を考慮すると，索引化が適用できる手法が有効であると考えられる[15)]．そこで本論文では，索引化技術を適用できる，ユークリッド距離を用いる手法[15)] と DP マッチングを用いる手法[11)] を対象にして提案手法と比較検討を行う．本提案手法の説明に移る前に，ユークリッド距離を用いた手法の概要を次節で説明する．

2.2 ユークリッド距離を用いた従来手法

音高推移特徴ベクトルを音楽特徴量に用い，ベクトル間のユークリッド距離を距離尺度に用いた検索方法を述べる[15)]．まず，検索対象の音楽データの主旋律をスライディング・ウインドウ方式により分割する．スライディング・ウインドウ方式とは，図 1 に示すように，固定ウインドウ長を「拍」を単位に，ある幅ずつスライドしてメロディ片に分割する方法である．スライド幅をウインドウ長より短くすることで，連続するメロディ片は互いに重なりのある冗長なデータとなり，検索する部分に関して自由度が増す．入力データも同様にスライディング・ウインドウ方式で分割する．

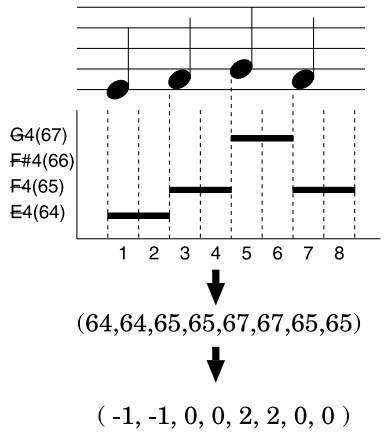


図 2 音高推移特徴ベクトルの生成例
Fig. 2 Example of a pitch-transition feature vector.

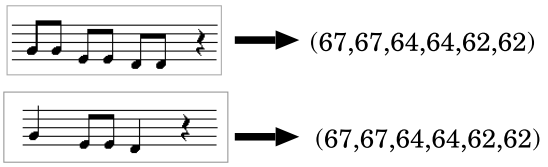


図 3 リズムの異なるメロディ片の音高推移特徴ベクトルの生成例
Fig. 3 Example of pitch-transition feature vectors for different melody data.

次に，各メロディ片の音高推移特徴ベクトルを作成する．音高推移特徴ベクトルの各要素は単位拍（拍粒度）ごとの音高と代表音（最も頻出する音）の音高との差で表現される．例として，**図 2** に音高推移特徴ベクトルの生成例を示す．図中の E4 や F4 は音コードを示し，その左の数字は MIDI の音高番号を示す．8 分音符を拍粒度とすると，このメロディからは 8 次元の特徴ベクトル (64,64,65,65,67,67,65,65) が作成される．このメロディでの代表音は最も頻出している F4(65) で，代表音との音高差を表現すると (−1,−1,0,0,2,2,0,0) というベクトルとなる．

音高推移特徴ベクトルは拍ベースの特徴量であるので音長が考慮されていない点，ハミング入力の曖昧さによって音高推移の代表音が変化すると特徴ベクトルのすべての値に影響を与える点の 2 つが問題となる．まず，拍ベースの問題点について述べる．**図 3** は出現する音符の音程は同じであるが，リズムの異なる 2 つのメロディ片から生成された特徴ベクトルである．拍ベースで特徴ベクトルを生成する場合，音長を考慮していないため，特徴ベクトル間のユークリッド距離が 0 となり，リズムの異なるメロディが同じメロディと

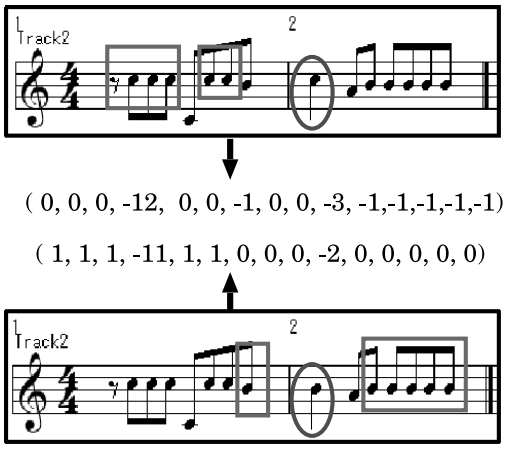


図 4 代表音の異なるメロディ片の音高推移特徴ベクトルの生成例
Fig. 4 Example of pitch-transition feature vectors for different base notes.

判断されてしまう．リズムの特徴を表すリズムベクトル[4] も提案されているが，固定長のメロディ片ではリズムベクトルの次元数が異なるため，ユークリッド距離を適用することができない．

次に，代表音の問題点について述べる．**図 4** は代表音が異なる音高推移特徴ベクトル生成例である．一見，2 つのメロディは同じメロディのように思えるが，○で囲んでいる 1 音符の音程がずれたために，代表音（□で囲んだ音符）が変化し，異なったベクトルが生成される．ハミングのように記憶を頼りに歌う場合，楽譜どおりに正しく歌うことはできない．音高推移特徴ベクトルを用いると，ほとんど同じメロディでも少しの音程のずれで異なる代表音が選択されるため，異なったベクトルが生成され，検索精度に影響を与える．

3. EMD を用いた類似音楽検索手法

3.1 EMD とは

EMD は線形計画問題の 1 つである輸送問題の解に基づいて計算される．輸送問題とは一定の供給量を持つ複数の供給地と同じく一定の需要量を必要とする需要地を設定し，各供給地から需要地までの輸送コストが与えられた際に，需要地の需要を満たすよう供給地から需要地へ最小の輸送コストで荷物を輸送する輸送方法を探す問題である．以下，EMD の計算方法を説明する．まず，m 個の供給地を持つ供給地集合 $\boldsymbol{P}$ と n 個の需要地を持つ需要地集合 $\boldsymbol{Q}$ をそれぞれ以下のように表す．

$$\boldsymbol{P} = \{(\boldsymbol{p}_1, w_{p_1}), ..., (\boldsymbol{p}_m, w_{p_m})\} \quad (1)$$

$$\boldsymbol{Q} = \{(\boldsymbol{q}_1, w_{q_1}), ..., (\boldsymbol{q}_n, w_{q_n})\} \quad (2)$$

ここで，$\boldsymbol{p}_i$ は i 番目の供給地を表す特徴ベクトル，w_{p_i} は i 番目の供給地が有する供給量，$\boldsymbol{q}_j$ は j 番目の需要地を表す特徴ベクトル，w_{q_j} は j 番目の需要地が必要とする需要量を示す．各供給地 $\boldsymbol{p}_i$ と各需要地 $\boldsymbol{q}_j$ 間の単位輸送あたりの輸送コスト（d_{ij}）を定義する．

$$d_{ij} = ||\boldsymbol{p}_i - \boldsymbol{q}_j|| \quad (3)$$

一般に，輸送コストとして各ベクトル要素 $\boldsymbol{p}_i$ と $\boldsymbol{q}_j$ のユークリッド距離を用いる．次に，$\boldsymbol{p}_i$ と $\boldsymbol{q}_j$ のすべての組合せを考慮し，総輸送コストを計算する．総輸送コストは，$\boldsymbol{P}$ から $\boldsymbol{Q}$ への輸送量（フロー：$F = \{f_{ij}\}$）を決定する以下の輸送問題の解を用いて計算する．任意の供給地・需要地の組合せによる総輸送コスト（WORK）は次式のように表すことができる．

$$\mathrm{WORK}(\boldsymbol{P}, \boldsymbol{Q}, F) = \sum_{i=1}^{m}\sum_{j=1}^{n} d_{ij} f_{ij} \quad (4)$$

ただし，総輸送コストを計算する場合，以下の制約条件（式 (5)〜(8)）を満たすものとする．

- 制約条件 1：供給地から需要地の 1 方向にしか輸送されない．

$$f_{ij} \geq 0, \ (1 \leq i \leq m, 1 \leq j \leq n) \quad (5)$$

- 制約条件 2：供給地 i から供給できる容量は供給量 w_{p_i} を超過しない．

$$\sum_{j=1}^{n} f_{ij} \leq w_{p_i}, \quad (1 \leq i \leq m) \quad (6)$$

- 制約条件 3：需要地 j が受け取れる容量は需要量 w_{q_j} 以下である．

$$\sum_{i=1}^{m} f_{ij} \leq w_{q_j}, \quad (1 \leq j \leq n) \quad (7)$$

- 制約条件 4：供給地から移動できる最大総輸送量

$$\sum_{i=1}^{m}\sum_{j=1}^{n} f_{ij} = \min\left(\sum_{i=1}^{m} w_{p_i}, \sum_{j=1}^{n} w_{q_j}\right) \quad (8)$$

最終的に EMD($\boldsymbol{P}$,$\boldsymbol{Q}$) は，上の輸送問題の最適値（総輸送コストの最小値）min(WORK($\boldsymbol{P}$,$\boldsymbol{Q}$,F)) を総フローで割って以下のように求まる．

$$\mathrm{EMD}(\boldsymbol{P}, \boldsymbol{Q}) = \frac{\min(\mathrm{WORK}(\boldsymbol{P}, \boldsymbol{Q}, F))}{\sum_{i=1}^{m}\sum_{j=1}^{n} f_{ij}} \quad (9)$$

EMD の計算処理の例を**図 5** に示す．図 5 において 2 台のトラックを供給地，3 個の□を需要地とし，それぞれ図に示す資源が割り当てられている．なお，供給量，需要量の総数はどちらも 10 個で同数である．また，供給地から需要地への経路は矢印で示され，矢印上の値がその経路での輸送コストを表す．このよう

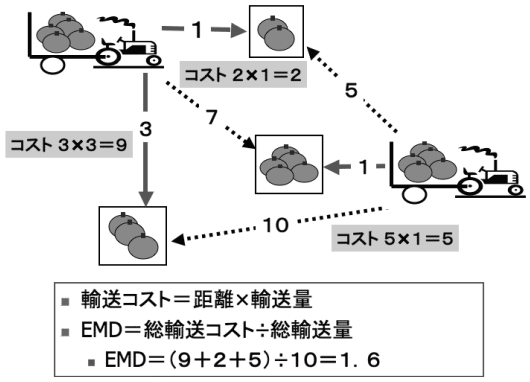


図 5 EMD の計算例
Fig. 5 Example of calculation of EMD.

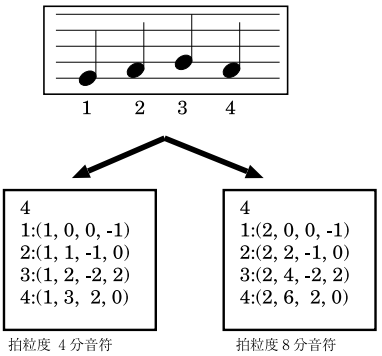


図 6 EMD に適用する音楽特徴量生成例
Fig. 6 Example of music features for EMD.

な条件下で，実線矢印の輸送経路が選択された場合，図 5 の下方に示す計算式に従って値が求まる．

距離尺度に EMD を用いた画像検索手法[21] では，色情報に基づいて画像を領域分割し，領域ごとに領域の広さ，平均色，重心点，テクスチャ情報を抽出する．そして，画像内の領域数を供給地数，各領域の広さを各供給地が有する供給量と見なす．さらに，各領域の平均色，重心点，テクスチャ情報を固定長のベクトルで表現し，各ベクトル間のユークリッド距離を輸送コストと見なす．そして，比較対象の 2 つの画像の一方を供給サイド，他方を需要サイドと見なして画像間の類似度を EMD によって計算する．EMD を適用した類似画像検索手法は，色合いのみでなく構図も考慮した画像を検索可能であり，高い検索精度が示されている．

3.2 EMD を用いた類似音楽検索手法

EMD を用いた類似画像検索が各領域の広さと各領域の画像的特徴の類似性を同じ距離尺度で判定できたように，類似音楽検索の距離尺度に EMD を適用すると，各音符の長さと各音符の音楽的特徴（出現時間，音高）の類似性を同じ距離尺度で判定でき，その結果，人間の感覚と同じように，全体の曲調が類似した曲を精度良く検索できるのではないかと我々は考えた．

EMD を類似音楽検索手法に適用するにあたり，スライディング・ウインドウ方式[15] により分割されたメロディ片ごとに音楽特徴量を作成する．これらのメロディ片ごとに音楽特徴量を作成することによりインデキシングが可能になる．また，メロディ片間の類似性はメロディ片内の音符を音長や音高が類似した他のメロディ片内の音符にマッピングするコストと考えられる．しかし，メロディ片内の音符数が一定でないため，時系列データである音符を分割してマッピングするコストの設定が必要である．このコストの計算に EMD を用いた場合，輸送問題における資源分配を音符の分割マッピングと同等に扱えるため，メロディ片内の音符数を供給地数，各音符の長さを各供給地が有する供給量と見なす．音符間の単位輸送コストを求めるための特徴量は，各音符の出現時間，前音との音高差，音高推移情報を固定長のベクトルで表現し，各ベクトル間のユークリッド距離を輸送コストとする．出現時間と音長は拍数で表現し，出現時間はウィンドウ内で当該音符が出現した拍数を表す．音高差は半音高い音を 1，半音低い音を -1 とする．

図 6 に 4 分音符を 1 拍とした場合と 8 分音符を 1 拍とした場合の音楽特徴量の例を示す．1 行目はメロディ片内の音符の数を表記し，2 行目以降は，出現音符に対する特徴量を表している．4 分音符を 1 拍とした場合，第 1 音は，

(音長，出現時間，音高差，音高推移) $= (1, 0, 0, -1)$

となり，8 分音符を 1 拍とした場合は，

(音長，出現時間，音高差，音高推移) $= (2, 0, 0, -1)$

となる．

このような音楽特徴量を採用することで，音程のずれに対しては，音符の音高情報が反映された輸送コストにより EMD が計算される．また，ハミングデータで頻出する音符の誤挿入に対しては，音符の出現時間といった時系列情報が反映された輸送コストに従って資源分配が行われ，EMD が計算される．特に，極端に音高の外れた音符（外れ音符）の誤挿入に対しては，DP マッチングを用いた場合，外れ音符の直前に位置する音符との音高差がそのままコストに反映されるため，外れ音符の有無が距離に大きな影響を与える．一方，EMD を用いた場合，外れ音符の周辺（前後）に位置する音楽特徴量が類似した複数の音符に資源配分のフローが作成され，最適なコストが計算されるため，外れ音符に対するノイズ軽減が期待できる．さらに，輸送コストをユークリッド距離で計算する際に，音符

の出現時間にパラメータ的な重みを与えることで，時間軸方向の制約を調整できる（パラメータを大きくすることで，時間軸方向の制約を強めた最適な資源分配方法が決定され，逆に弱くすれば，時間軸方向の制約が弱まる）．以上のように，音楽特徴量間の類似性をEMD を用いて判定することにより，ハミング検索で重要な課題になる音程のずれやリズムの違いに対して柔軟な検索が可能になる．

4. 検索処理の高速化への改良

4.1 EMD を用いた検索処理の問題点

EMD を用いた検索処理では，検索速度が遅いことが実用化への問題となる．検索速度が遅い理由としては，以下の 2 点があげられる．

距離計算回数が多い：

曲をメロディ片に分割しているため，1 曲が複数のメロディ片に分割される．その結果，メロディ片数は膨大になり，すべてのメロディ片に対して距離計算を行うと，多大な計算時間が必要となる．

時間計算量が大きい：

EMD の計算には一般にシンプレックス法が用いられるため，音符数を N とすると，N に対して指数関数的に時間計算量が増加する．メロディ片内には音符数が多く☆，多大な計算時間が必要となる．

4.2 改良手法

まず,「距離計算回数が多い」といった問題点に対しては，音楽データベースの索引化手法が適用できる．音高推移特徴ベクトルを用いた従来手法[15] では，ユークリッド空間内のデータを索引化する SR-tree[22] を採用している．一方，EMD はユークリッド空間では表現されないため，同様の索引化手法は適用できない．そこで本手法では，EMD が距離の 3 公理[23] を満たす点に着目し，距離空間内のデータを索引化する VP-tree[24] を採用し，検索速度の向上を図っている．

次に,「時間計算量が大きい」といった問題点に対しては，音楽特徴量内の音符数を削減する必要がある．音楽特徴量内の音符数を削減させるためには，メロディ片の長さを単に短くする方法が考えられる．しかし，メロディ片を短くすると情報量が減少し，検索精度が低下することが容易に想像できる．そこで本手法では，メロディ片の長さを変化させずに（メロディ片内の情報量を減少させずに），音楽特徴量内の音符数だけを削減させる改良を行った．本改良手法では，**図 7** に示すように，メロディ片の長さを変えずにウィンドウ内で偶数番目に出現する音符（スキップ音符）の特徴量を削除し，スキップ音符の音長を前の音符（基準音符）の音長に加算する．さらに，スキップ音符が本来存在しなかったメロディ片と区別するため，輸送コストを求める特徴量に基準音符間の音高差（2 つ前の音符との音高差）を加える．最終的に各音符の特徴量を以下のように変更し，スキップ音符の特徴を補う音楽特徴量を作成する．

（基準音符の音長＋スキップ音符の音長，基準音符の出現時間，基準音符とスキップ音符間の音高差，基準音符の音高推移，基準音符間の音高差）

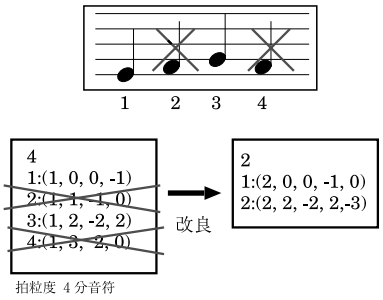


図 7 特徴量の改良方法
Fig. 7 Improvement of music features.

5. 評　　価

5.1 実験条件

検索対象の音楽データベースとして，童謡，J-pop，演歌等のジャンルが含まれるカラオケ用 MIDI 音楽データ 483 曲を使用した．これら市販の MIDI 音楽データは，特定のチャネルに主旋律が格納されているため，機械的に主旋律のデータのみを自動抽出した．その後，主旋律のデータに対して，スライディング・ウインドウ方式を適用して，メロディ片を生成した．スライディング・ウインドウの条件としては，8 分音符を 1 拍とし，ウインドウ長 16 拍，スライド長 4 拍として 84,554 個のメロディ片を生成した．メロディ片内の各音符に対して 8 分音符の長さを 1 とした音長，出現時間を用いて音楽特徴量を作成し，VP-tree[24] を用いて音楽特徴量の索引化を行った．検索入力には，男女 6 名が歌ったハミングを採用し，市販の採譜ソフト[25] で MIDI 形式に変換した 40 曲を用いた．採譜方法としては，ハミングの長さが最低でもウインドウ長を超える条件を義務付け，ハミングの際には，正確なハミングは要求せず，その曲を知っている人が聞いて分かるレベルの入力とした．そのため，入力ハミン

☆ 今回の実験データでは，2 小節のメロディ片内での平均音符数は約 14 であった．

グにはリズム，音程のずれが生じた．また，検索結果の順位付けの方法は以下の手順に従った．まず，検索対象の音楽データと同じ条件下でハミングをメロディ片に分割し，分割した 693 個のハミングメロディ片ごとに上位 50 件の検索結果を得る．その後，すべてのハミングメロディ片の検索結果を統合し，類似度順にソートした結果を検索結果とした．よって，目的の曲と類似したハミングメロディ片が入力ハミング内に 1 つでも存在していれば，検索結果の上位に目的の曲がリストアップされる☆．なお，実験に用いた計算機の OS は Linux，CPU は Pentium-4 の 2.4 GHz，メモリは 512 MB であった．

5.2 ユークリッド距離との検索精度の比較評価

ユークリッド距離を用いた従来手法[15] として，メロディ片ごとに音高推移ベクトルを生成し，ユークリッド距離でベクトル間の類似度を計算した．**図 8** に提案手法と従来手法との検索精度を示す．横軸は検索結果の順位，縦軸はその順位以内に正解のメロディ片が検出された曲数の割合（正解率）を表す[15]．たとえば，5 位以内で 80%の正解率の場合，入力ハミング 40 曲中 32 曲において 5 位以内に正解が含まれていたことを意味する．また，提案手法に関しては，輸送コストを計算する音符の特徴量を変化させた結果も示す．

実験結果から EMD を用いることで，従来手法より検索精度が向上したことが分かる．提案手法において，輸送コストを音符の出現時間のみで計算した場合，リズムしか考慮されないため，検索精度が低い結果となった．出現時間に音高情報を加えることでリズムと音程が同時に考慮され，検索精度が大幅に向上している．音高情報に関しては，音高差だけではなく，音高推移を加えたものが最も精度が高かった．提案手法で最も高い精度が得られた結果と従来手法を比較すると，5 位以内で 27%，10 位以内では 30%の正解率の向上が見受けられ，順位を増すに従って，正解率の向上が大きくなった．また，入力ハミングのすべてについて 30 位以内に正解が含まれていた．

次に，提案手法を用いることで検索結果が向上した原因を考察するため，提案手法では 1 位に正解データが検索され，従来手法では 1 位に検索されなかったデータを分析する．まず，入力ハミングメロディ片 693 件のうち，EMD による検索で 1 位に正解データが検索されたものは 161 件存在した．この中でユークリッド距離による検索でも 1 位に検索されたものは 81 件であったため，残りの 80 件のハミングメロディ片と対応する正解データに対して，それぞれの音高推移ベクトルを作成し，ユークリッド距離で検索精度が低下した原因を分析した．

☆ 検索結果を曲ごとにマージする場合は，ハミングメロディ片に対する距離の逆数を類似度とし，曲ごとの類似度の総数をソートする．

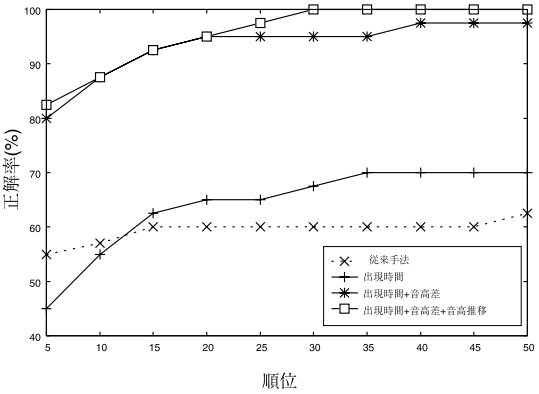


図 8 検索精度の比較実験結果（ユークリッド距離）
Fig. 8 Experimental result of retrieval precisions.

表 1 ユークリッド距離における検索精度低下の分類
Table 1 Classification of retrieval results by Euclid distance.

分類項目	件数	比率 (%)
代表音のずれ	20	25.0
ウィンドウのずれ	30	37.5
音長のずれ	36	45.0
音程のずれ	47	58.8

表 1 に分析結果を示す．分類項目の中で「代表音のずれ」と「音長のずれ」は，2.2 節で説明した問題点を表す．「ウィンドウのずれ」とは，ハミングの歌い出し位置が任意であるので，ハミングメロディ片と同じ位置で切り出されたメロディ片がデータベース内に必ずしも存在しないことから生じる問題点であり，文献 16) でも指摘されている．「音程のずれ」とは，代表音は正しいが，その他の音符の音高がずれていることを意味する．件数は各分類項目に該当したハミングメロディ片の件数を表し，比率は各分類項目の件数が調査データ全体（80 件）に占める割合を示す．なお，1 つのハミングメロディ片に該当する項目が複数あるため，件数の合計が調査データ全体数を超過している．

表 1 より，約半数のデータで「音長のずれ」が見受けられ，音高推移ベクトルの問題点とすべき正当性が確認できる．「ウィンドウのずれ」に対しては，データベース構築時に，ウィンドウのスライド長を小さくし，より細かくメロディ片を切り出すことも可能だが，データベースのサイズが大きくなり，検索精度と速度の両面に悪影響を与える．この問題点に対して EMD

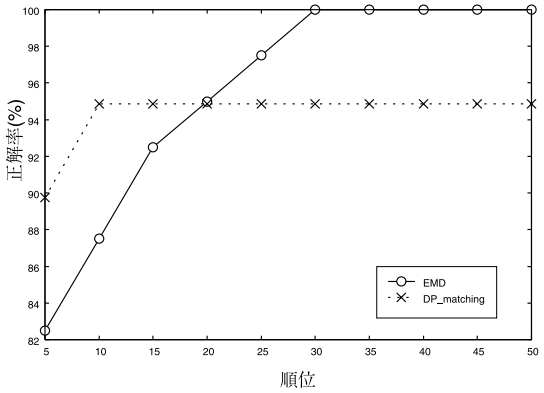


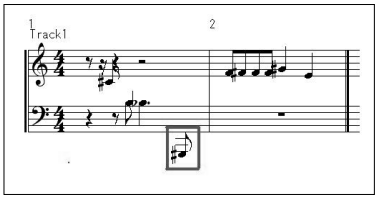
図 9 検索精度の比較実験結果（DP マッチング）
Fig. 9 Experimental result of retrieval precisions.

を用いた場合，時間軸方向に各音符をスライドして最適なコストを計算したため，正解データを 1 位に検索できたのだと考えられる．なお，ユークリッド距離を用いた検索は，検索時間の高速化を目的に考案された手法である．今回用いた実験データで計測したところ，1 ハミングメロディ片あたり 0.07 秒で検索が行え，検索時間の面では本手法を大幅に上回っている．

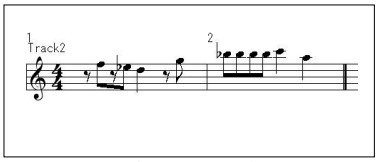
5.3 DP マッチングとの検索精度の比較評価

DP マッチングを用いた従来手法[11] として，音長と音高差の距離の総和を用いて DP マッチングを行い類似度を計算した．ただし，ハミング系列全体を用いるのが通常の DP マッチングであるが，文献 11) に従ってハミングメロディ片ごとにデータベース内のメロディ片との DP マッチングを行った．このような処置を行うのは，索引化を可能にするためだが，比較系列数が短くなるため，通常の DP マッチングと比べて検索精度が若干低下する．なお，検索結果の順位付けの方法は本手法と同様にした．**図 9** に提案手法と従来手法との検索精度を示す．縦横軸の内容は，図 8 と同じである．

実験結果より，検索結果上位 20 以内では DP マッチングの方が提案手法より若干高い精度を残した．しかし，提案手法では検索結果上位 30 以内にすべての正解データを検索できたのに対して，DP マッチングで 50 位以内に検索できなかったハミングデータが 2 曲存在した．その 2 曲を分析したところ，**図 10**，**図 11** に示すように，極端に音高の外れた音符（外れ音符）を 2 曲とも含んでいた☆．このような外れ音符が含まれていた場合，DP マッチングでは外れ音符と正解音符との差がそのままコストに反映され，正解データとの距離が大きくなる．入力系列数（音符数）が長ければ，少々の外れ音符が出現してもさほどコストに影響を与えないが，テンポがスローなハミングや今回のようなハミングメロディ片に対する系列の場合，音符数が少ないため，外れ音符の影響が大きくなったと考えられる．一方，提案手法を用いた場合は，時間軸方向の制約が DP マッチングほど強くないため，外れ音符の周辺（前後）に位置する音楽特徴量が類似した複数の音符にフローが作成され，正解データとの距離の増加が抑えられたと考えられる．

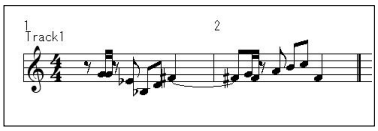


入力ハミングメロディ片

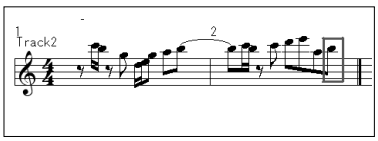


正解 のメロディ片

図 10 外れ音を含むハミングメロディ片の例（その 1）
Fig. 10 Example of the humming submelody including outlier.



入力ハミングメロディ片



正解 のメロディ片

図 11 外れ音を含むハミングメロディ片の例（その 2）
Fig. 11 Example of the humming submelody including outlier.

このように提案手法では，DP マッチングに比べて時間軸方向の規制が強くないが，輸送コストを計算する際に音符の出現時間にパラメータ的な重みを与えることで，時間軸方向の制約を調整できるといった長所がある．音符の出現時間に対する重みパラメータを 1 倍（1times），2 倍（2times），3 倍（3times）と変化させた際の検索精度の変化を**図 12** に示す．実験結果より，パラメータを変化させることで出現時間に対す

☆ 図 11 ではハミング側に正解データの最終音符が脱落しており，脱落に対する音高差距離が 8（ハミング側が 6，正解データ側が −2）になることで全体のコストが増加していた．

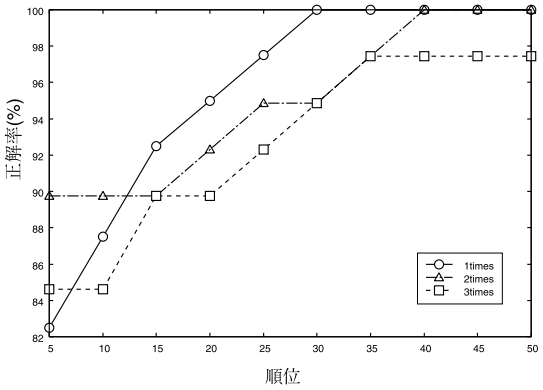


図 12 出現時間の重みを変化させた検索精度
Fig. 12 Relationship between times parameter and retrieval precisions.

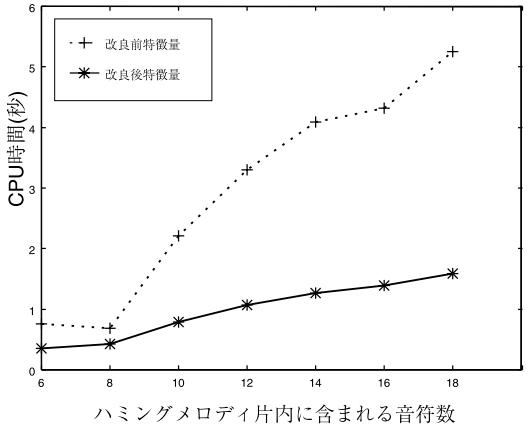


図 13 音符数ごとの検索時間
Fig. 13 CPU time every number of notes on the retrieval process.

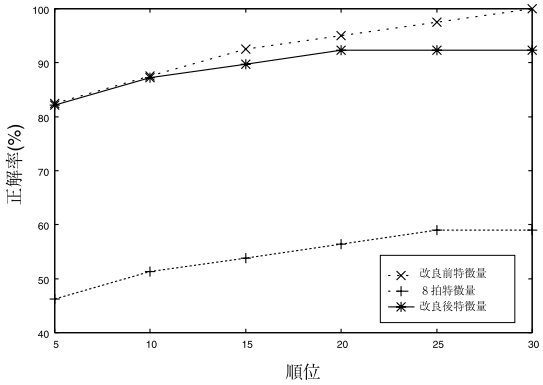


図 14 特徴量改良後の検索精度
Fig. 14 Retrieval precision on the improved method.

る制約が変化し，検索特性が変動することがうかがえる．また，パラメータの重みを 2 倍にした結果が図 9 に示した DP マッチングの精度に近づいていることが分かる．このように，EMD をハミング検索に適用すると，ハミングデータに含まれるノイズに従ってパラメータ値を調整することで，より柔軟な検索が可能になる．

5.4 検索速度の改良手法に対する評価

本論文で提案した検索速度の改良手法は，ハミングメロディ片内に含まれる音符数に着目したものである．そこで，ハミングメロディ片内に含まれる音符数に従って検索速度がどの程度改善したかを確認するため，ハミングメロディ片の中に含まれる音符数ごとに各ハミングメロディ片を分類し，出現音符数ごとのハミングメロディ片に対して平均検索時間を計測した．**図 13** は，改良手法の適用前後における検索速度の比較実験結果である．横軸はハミングメロディ片に含まれる音符数，縦軸は対応するハミングメロディ片に対する検索処理に要した CPU 時間の平均を表す．図中の実線が改良後の特徴量を用いた検索時間，破線が改良前の検索時間である．図 13 より，改良前の特徴量では音符数の増加に従って検索時間が指数関数的に増加しているが，改良後の特徴量では検索時間の増加が抑えられていることが分かる．平均音符数 14 においては，約 4 倍の高速化に成功している．

次に，改良手法の適用前後における検索精度の比較実験結果を**図 14** に示す．縦横軸の内容は図 8 と同じである．図 14 より，改良後の特徴量は改良前に比べて，若干精度が低下しているが，十分にスキップ音符を補っていることが分かる．特に，元々上位に検索されていた精度の良いハミングに対しては，検索精度の低下は見られなかった．なお，単純に音符数を削減した特徴量を作成する方法として，ハミングメロディ片のウィンドウ長を狭める方法が考えられる．そこで，ウィンドウ長を半分の 8 拍にした改良前の特徴量に対する検索精度も図 14 に示す．8 拍の特徴量では情報量が少なくなったため，検索精度が大幅に低下した．図 13 と図 14 の結果から，検索精度と検索時間の両方を優先するのであれば，改良手法が適しているといえる．ただし，今回は小規模なデータベースに対する実験であるため，実用化レベルの大規模なデータベースに適用すると，さらなる精度低下が発生する可能性もある．そのため，大規模なデータベースに提案手法を適用し，検索精度および速度評価の両面においてさらなる検証と考察が必要である．なお，今回は音楽特徴量の索引化に VP-tree を用いたが，拡張版の VP-tree[26] も提案されており，索引化の面からもさらなる高速化が期待できる．

5.5 類似研究との比較

文献 27) では，本手法と同じように EMD を用いた類似音楽検索手法を提案しており，クラッシック音楽の自動分類を行っている．これに対して，本研究では入力に誤りを含むハミング検索へ EMD を適用しており，適用事例の違いから特徴量の設定方法にも相違点がある．つまり，先行研究では音高情報として絶対音高を用いているが，前音との音高差や音高推移情報を用いている点，音符の出現時間に重みパラメータを設定している点等が異なる．また，時間効率を改善する工夫について先行研究ではまったく触れられておらず，提案手法の新規性は高い．さらに，先行研究では DP マッチング等の従来手法との比較実験も行われておらず，EMD を音楽検索に適用した際の有用性を客観的に議論できていない．これに対して本研究では，ユークリッド距離および DP マッチングとの比較実験から EMD をハミング検索に適用した際に生じる有用性を発見しており，本研究の優位性は高いと考えられる．

6. ま と め

本論文では，距離尺度として EMD を用いたハミングによる類似音楽検索手法を提案した．さらに，EMD を適用する音楽特徴量に対して，音符数を削減する改良手法を提案した．実際のハミングデータを用いた評価実験では，ユークリッド距離を用いた従来手法に比べて検索精度を向上させることができ，DP マッチングを用いた手法に比べて時間軸方向の制約を柔軟に調整できることが確認できた．また，音楽特徴量の改良手法を適用することによって，検索精度を維持しつつ，検索処理の高速化が実現できた．

今後の課題としては，実用化レベルの大規模なデータベースへの適用があげられる．大規模なデータベースに対しては，検索処理の高速化が必須の課題になると考えられるので，新たな索引化手法の考案，データベースの冗長性の削減等に対して取り組みたい．また，今回行った DP マッチングとの比較実験もデータ数が少なかったため，より大量のハミングデータと比較し，本手法の頑健性を検証する予定である．

謝辞 本研究の一部は，科学研究費補助金基盤研究（B）（17300036），科学研究費補助金基盤研究（C）（17500644）を受けて行われた．

参 考 文 献

1) 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).
2) 帆足啓一郎，上月勝博，菅谷史昭：楽曲配信サービスを支える音楽情報検索技術，電子情報通信学会誌，Vol.88, No.7, pp.529–534 (2005).
3) 武田晴登，篠田浩一，嵯峨山茂樹：リズムベクトルを用いたリズム認識，情報処理学会音楽情報科学研究会資料，MUS-46-4, pp.23–28 (2002).
4) 池谷直紀，服部正典，大須賀昭彦：リズム入力による音楽検索方式「タタタタップ」，第 3 回情報科学技術フォーラム，No.G-021, pp.391–393 (2004).
5) Dannenberg, R.B. and Hu, N.: Understanding Search Performance in Query-By-Humming Systems, *Proc. 5th International Symposium on Music Information Retrieval*, pp.232–237 (2004).
6) Colin, M. and William, B.: Johnny Can't Sing: A Comprehensive Error Model for Sung Music Queries, *Proc. 3rd International Symposium on Music Information Retrieval*, pp.124–132 (2002).
7) Jyh-Shing, R.J., Hsu, C.L. and Lee, H.R.: Continuous HMM and Its Enhancement for Singing/Humming Query Retrieval, *Proc. 6th International Symposium on Music Information Retrieval*, pp.546–551 (2005).
8) Ito, A., Heo, S.P., Suzuki, M. and Makino, S.: Comparison of Features for DP-Matching Based Query-By-Humming System, *Proc. 5th International Symposium on Music Information Retrieval*, pp.297–302 (2004).
9) Parker, C.: Applications of Binary Classification and Adaptive Boosting to the Query-By-Humming Problem, *Proc. 6th International Symposium on Music Information Retrieval*, pp.245–251 (2005).
10) Steffen, P.: CubyHum: A Fully Operational Query By Humming System, *Proc. 3rd International Symposium on Music Information Retrieval*, pp.187–196 (2002).
11) Sonoda, T. and Muraoka, Y.: A WWW-based Melody-Retrieval System—An Indexing Method for A Large Melody Database, *Proc. ICMC 2000*, pp.170–173 (2000).
12) 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，電子情報通信学会論文誌，Vol.J82-D-II, No.4, pp.721–731 (1999).
13) 西村拓一，滝田順子，後藤真孝，岡 隆一：類似メロディー区間検出による音楽時系列検索の高速化，情報処理学会音楽情報科学研究会資料，MUS-39-10, pp.63–70 (2001).
14) 許 盛弼，鈴木基之，伊藤彰則，牧野正三：複数の音高候補値を用いた楽曲検索システムの構築，情報処理学会音楽情報科学研究会資料，MUS-49-15, pp.85–90 (2003).

15) 小杉尚子，小島　明，片岡良治，串間和彦：大規模音楽データベースのハミング検索システム，情報処理学会論文誌，Vol.43, No.2, pp.287–298 (2002).
16) 小杉尚子，櫻井保志，山室雅司，串間和彦：SoundCompass：ハミングによる音楽検索システム，情報処理学会論文誌，Vol.45, No.1, pp.333–345 (2004).
17) Dannenberg, R.B., William, P.B., George, T., Colin, M., Ning, H. and Bryan, P.: The MUSART Testbed for Query-By-Humming Evaluation, *Proc. 4th International Symposium on Music Information Retrieval*, pp.41–50 (2003).
18) 熊本忠彦，太田公子：印象に基づく楽曲検索研究のための印象表現の収集，情報処理学会論文誌，Vol.43, No.10, pp.3231–3234 (2002).
19) Hoashi, K., Matsumoto, K. and Inoue, N.: Personalization of user profiles for content-based music retrieval based on relevance feedback, *Proc.11th ACM International Conference on Multimedia*, pp.110–119 (2003).
20) Grachten, M., Arcos, J.L. and Mantaras, R.L.: Melodic Similarity: Looking for a Good Abstraction Level, *Proc. 5th International Symposium on Music Information Retrieval*, pp.210–215 (2004).
21) Rubner, Y., Tomasi, C. and Guibas, L.J.: The earth mover's distance, multi-dimensional scaling, and color-based image retrieval, *Proc. ARPA Image Understanding Workshop*, pp.661–668 (1999).
22) 片山紀生，佐藤真一：SR-tree：高次元点データに対する最近接検索のためのインデックス構造の提案，電子情報通信学会論文誌，Vol.J80-D-I, No.8, pp.703–717 (1997).
23) Yianilos, P.N.: Data structures and algorithms for nearest neighbor search in general metric spaces, *Proc. ACM-SIAM SODA'93*, pp.311–321 (1993).
24) Fu, A.W.-C., Chan, P.M.S., Cheung, Y.-L. and Moon, Y.S.: Dynamic vp-tree indexing for n-nearest neighbor search given pair-wise distances, *VLDB Journal*, pp.2–8 (2000).
25) 鼻歌ミュージシャン 2：株式会社メディア・ナビゲーション.
26) 中川嘉之，獅々堀正幹，北　研二：距離索引 VP-tree における解絞り込みの一改良法，情報処理学会情報学基礎研究会資料，FI-72-1, pp.1–8 (2003).
27) Typke, R., Giannopoulos, P., Veltkamp, R.C., Wiering, F. and Oostrum, R.: Using Transportation Distances for Measuring Melodic Similarity, *Proc. 4th International Symposium on Music Information Retrieval*, pp.107–114 (2003).

(平成 18 年 5 月 8 日受付)
(平成 18 年 10 月 3 日採録)

獅々堀正幹（正会員）

平成 3 年徳島大学工学部情報工学科卒業．平成 5 年同大学院博士前期課程修了．平成 7 年同大学院博士後期課程退学．同年同大学工学部知能情報工学科助手．平成 9 年同大学工学部知能情報工学科講師．平成 13 年同大学工学部知能情報工学科助教授．博士（工学）．マルチメディア情報検索，自然言語処理の研究に従事．著書『情報検索アルゴリズム』（共立出版），情報処理学会第 45 回全国大会奨励賞受賞．電子情報通信学会会員，言語処理学会会員．

大西　泰代（学生会員）

平成 17 年徳島大学工学部知能情報工学科卒業．現在，同大学大学院工学研究科博士前期課程知能情報工学専攻 2 年．情報検索の研究に従事．

柘植　　覚（正会員）

平成 8 年徳島大学工学部知能情報工学科卒業．平成 10 年同大学大学院工学研究科博士前期課程知能情報工学専攻修了．平成 13 年同大学院工学研究科博士後期課程システム工学専攻修了．平成 12 年徳島大学工学部助手．平成 18 年徳島大学工学部講師，博士（工学）．音声認識，情報検索等の研究に従事．日本音響学会会員．

北　研二（正会員）

昭和 56 年早稲田大学理工学部数学科卒業．昭和 58 年沖電気工業（株）入社．昭和 62 年 ATR 自動翻訳電話研究所出向．平成 4 年徳島大学工学部講師．平成 5 年同助教授．平成 12 年同教授．平成 14 年同大学高度情報化基盤センター教授．博士（工学）．自然言語処理，情報検索等の研究に従事．平成 6 年日本音響学会技術開発賞受賞．著書『確率的言語モデル』（東京大学出版会),『情報検索アルゴリズム』（共立出版）等．電子情報通信学会会員，言語処理学会会員．