音 楽 情 報 科 学　42－3
（２００１．１０．２６）

音楽検索のためのMIDIデータの近似マッチング法

高須 淳宏 †　　金澤 輝一 ‡　　安達 淳 †
† 国立情報学研究所
‡ 東京大学大学院 情報理工学系研究科
{takasu,tkana,adachi}@nii.ac.jp

音楽検索では問い合わせに特徴のあるフレーズが用いられることが多い。本稿では、ポピュラーミュージックを対象として、楽曲からフレーズを抽出し、特徴的なフレーズとしてサビを抽出する方法を述べる。提案手法は、楽曲のフレーズへの分割とフレーズの分類という２つのステップで構成されれている。楽曲のフレーズへの分割にはヒューリスティクな規則を用いており、フレーズの分類には、決定木と隠れマルコフモデルを組み合わせた確率構造解析技術を用いている。94 曲の楽曲に対して本手法を適用したところ、95.2%の分類性能を得た。

キーワード： 音楽検索、近似マッチング、構造解析、隠れマルコフモデル

Approximate Matching of MIDI Data for Music Retrieval

Atsuhiro Takasu†　　Teruhito Kanazawa‡　　Jun Adachi†
†National Institute of Informatics
‡School of Information Science and Technology, University of Tokyo

Theme phrase is often used in music retrieval. This paper proposes a phrase extraction and classification method to extract theme phrase for popular music. The proposed method consists of two steps: phrase extraction and syntactical classification of segmented fragments of melodies. Phrase extraction is carried out based on a few heuristic rules, while the syntactical classification is based on a probabilistic syntactical pattern analysis that combines decision tree classification and syntactical analysis using hidden Marcov model. We conducted an experiment on the accuracy of phrase classification using 94 Japanese popular songs and obtained 0.952 accuracy for correctly extracted phrases in theme classification.

Keywords: Music Retreival, Approximate Matching, Syntactical Analysis, Hidden Marcov Model

1　はじめに

音楽検索では、MIDI 楽器やハミングなどが問い合わせに用いられる。特にハミングを問い合わせに用いる場合には、ハミングにおける音程のずれやハミング認識における認識誤りが生じるため、検索時にデータベース中の楽曲との間で近似マッチングを行う必要がある（例えば、[2, 5]）。近似マッチングアルゴリズムに関しては、テキストを対象として多くの研究が行われており、その手法が音楽検索の近似マッチングにも応用されている。

一方、音楽検索においては、特徴的なフレーズが問い合わせに使われることが多い。そこで、音楽検索におけるランキングにおいて、楽曲の特徴的なフレーズとのマッチングを優先することによって検索性能の向上が期待される。そこで、本稿では、ポピュラーミュージックを対象として、特徴的なフレーズとしていわゆる「サビ」を抽出する方法を提案する。提案手法は、楽曲のフレーズへの分割と抽出されたフレーズの分類によって構成されている。本稿は、フレーズ分類を中心に提案手法を述べる。まず、次節で筆者らが試作してき

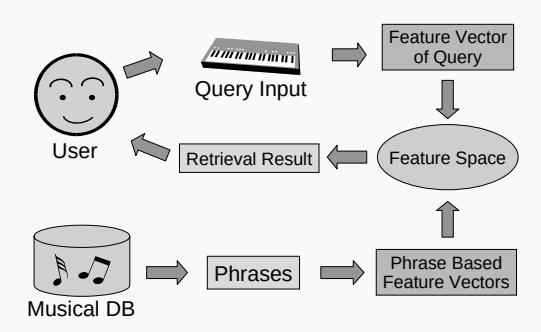

図 1: 検索処理の概要

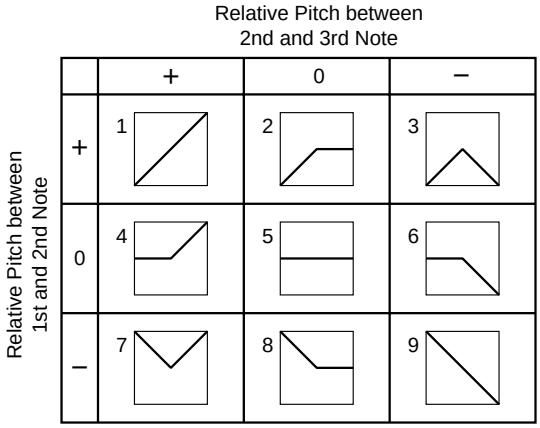

図 2: 相対音高列に対する特徴

た音楽検索システムの概要を示し、そのシステムにおける楽曲のフレーズ分割法を述べる。続いて 3 節で確率に基づいた構造解析によるフレーズ分類法を述べ、4 節で小規模音楽データを用いた実験結果を示す。

2 音楽検索システムの概要

著者等は、Musical Instrument Digital Interfance (MIDI) 形式で格納された音楽データベースに対する検索システムを試作してきた [6]。このシステムは、問い合わせとして、ハミングや MIDI 楽器による楽曲の部分演奏を問い合わせとして受取り、その問い合わせに類似したメロディを含むを楽曲を検索結果として提示するシステムとなっている。このシステムは、ポピュラーミュージックを対象とした音楽検索システムであり、楽曲のフレーズを用いることによって検索性能と検索効率を向上させるところに特徴がある。

このシステムは、まず、データベースに含まれる楽曲をフレーズに分割する。次に各フレーズの主旋律を相対的な音高列と音長列という 2 つのシークエンスデータと見なす。このシークエンスデータは、文字列マッチングで使われてきた n-gram 出現頻度を用いて特徴ベクトルに変換される [3]。問い合わせに使われるフレーズも同じ方法で特徴ベクトルに変換され、情報検索で用いられる内積による類似度を用いて検索を行う。図 1 は筆者らが試作した音楽検索システムにおける検索処理の流れを表している。

N-gram による特徴ベクトルを音楽検索に使うに際して、問い合わせにハミングを使う場合は、音程のずれがあるため、相対的な音高、音長列をそのまま使うことは望ましくない。そこで、相対音高を、前の音と比較し「高い」「等しい」「低い」の 3 状態の列として表すことによって、音程のずれを扱っている。このような特徴は文献 [5] で用いられたものと同じものである。図 2 に bigram に対する相対音高列（音長列も同様）の特徴を列挙する。フレーズの相対音高列中に各特徴が出現した頻度を特徴量として特徴ベクトルが構成される。bigram の場合は、音高列に対して 9 次元の特徴ベクトルが構成され、音長列と併せて、各フレーズは 18 次元の特徴ベクトルで表される。問い合わせも同じ方法で特徴ベクトル化され、内積を用いた類似度が計算される。

ポピュラーミュージックは、いわゆる「A メロ」、「B メロ」、「サビ」といったフレーズで構成されており、検索においてもこれらのフレーズが基本的な単位となると考えられる。これらのフレーズを抽出するために、我々はフレーズを構成する単位としてサブメロディを用いた。サブメロディは、フレーズを構成する音符列のなかで休符で挟まれた部分音符列を表す。ポピュラーミュージックにおいては、楽曲中に同種のフレーズが繰り返されるので、この類似性を測るための最小部分音符列としてサブメロディを用いている。図 3 は、本システムで想定しているポピュラーミュージックの階層構造を示している。

フレーズは、以下の手順で抽出される。

1. サブメロディの切出
2. サブメロディのクラスタリング

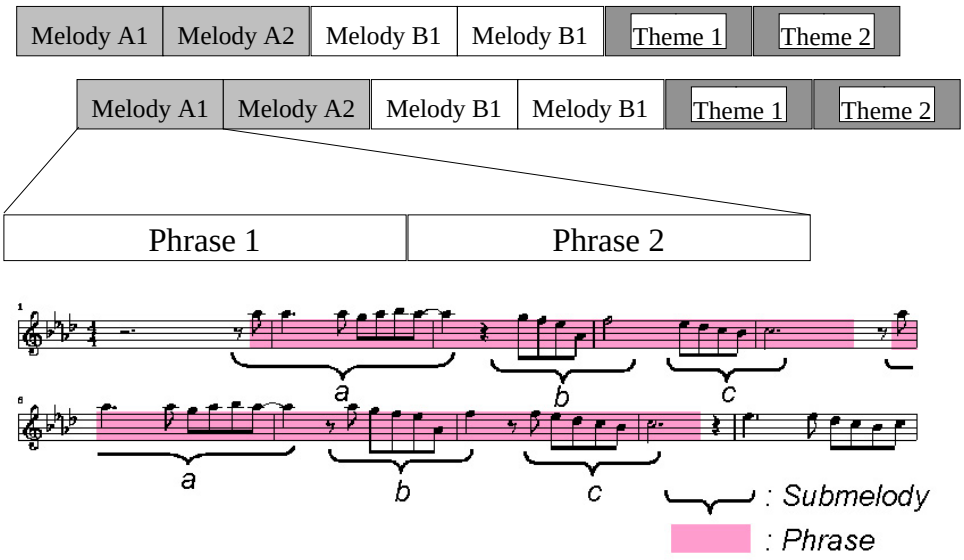

図 3: 楽曲のフレーズ構造

3. prefix cluster の抽出

ここで、prefix cluster とは、フレーズの先頭に位置するサブメロディからなるクラスタのことで、これにより楽曲がフレーズに分割される。

ステップ 1 は、MIDI データに対して機械的に実現することができる。ステップ 2 では、クラスタリングのために、まず、類似度を定義する必要がある。筆者らのシステムでは、音高列に対する最長一致文字列長に基づいて計算される。2 つのサブメロディの音高列を

$$\begin{aligned} S_1 &= a_1, a_2, \cdots, a_n \\ S_2 &= b_1, b_2, \cdots, b_m \end{aligned}$$

とした場合に、各サブメロディの位置 i, j に対して最長一致音高列長 $s(i,j)$ を以下のように定義し、

$$\begin{aligned} s(0,0) &= 0 \\ s(i,0) &= s(i-1,0) \\ s(0,j) &= s(0,j-1) \\ s(i,j) &= max \left\{ \begin{array}{l} s(i-1,j) \\ s(i-1,j-1)+1 \\ s(i,j-1) \end{array} \right\} \end{aligned}$$

2 つのサブメロディの類似度を

$$\frac{s(n,m)}{\max\{n,m\}}$$

と定義する。

上記の類似度に従って、サブメロディ間の類似度の分布をみてみると、同一クラスタ内のサブメロディの類似度の分布と異なるクラスタに属するサブメロディ間の類似度には、ギャップが存在することが多い。そこで、同一クラスタ内のサブメロディ間の類似度と異なるクラスタに属するサブメロディ間の類似度はそれぞれ異なる正規分布に基づいて分布すると仮定して、任意のサブメロディ間の類似度分布として以下の式で表される混合正規分布用いる。

$$\lambda N(\mu_1, \sigma_1^2) + (1-\lambda) N(\mu_2, \sigma_2^2)$$

ここで、$N(\mu, \sigma^2)$ は、平均 μ、分散 σ^2 の正規分布を表している。μ_1 と σ_1^2 は異なるクラスタに属するサブメロディ間の類似度の平均と分散を表し、μ_2 と σ_2^2 は同一クラスタに属するサブメロディ間の類似度の平均と分散を表す。また、λ は、2 つのサブメロディが異なるクラスタに属する確率を表している。各楽曲ごとに、パラメタ $\mu_1, \mu_2, \sigma_1^2, \sigma_2^2, \lambda$ を最尤推定し、$N(\mu_1, \sigma_1^2) = N(\mu_2, \sigma_2^2)$ となる類似度を閾値として、クラスタリングを行う。

ステップ 3 の prefix cluster の抽出では、以下に示すヒューリスティクスを用いた。

1. 楽曲の先頭にあるサブメロディを含むクラスタ
2. 直前に 4 拍以上の長さの休符があるサブメロディを含むクラスタ
3. 繰り返し現れるサブメロディ列の先頭に位置

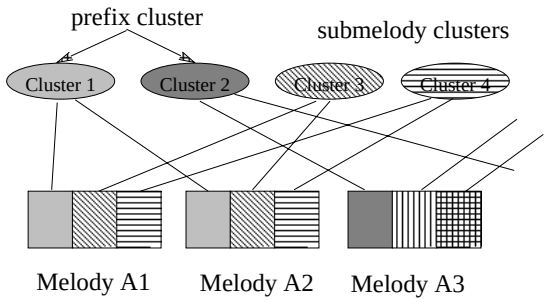

図 4: サブメロディのクラスタリング

するサブメロディを含むクラスタ

上記の処理によって得られたフレーズを、相対音高、音長列の n-gram 出現頻度に基づいて特徴ベクトル化を行い、音楽検索実験を行ったところ、楽曲全体を特徴ベクトル化するよりも検索性能が向上することが実験的に示された [4]。

3 フレーズの分類

ポピューラミュージックにおいては、問い合わせに使われるのはサビであることが多く、また、検索結果の提示においてはサビは適切なフレーズとなる。そこで、音楽検索においては、フレーズを分類し、サビを抽出することによってさらに検索性能の向上が期待できる。

フレーズ分類おいては、フレーズ個々の特徴と楽曲中におけるフレーズの位置を考慮する必要がある。そこで、本研究では、フレーズの特徴とフレーズ列の構造解析を組み合わせたフレーズ分類法を用いた。フレーズのカテゴリの集合を C とし、特徴ベクトル列で表されるフレーズ列 $\mathbf{v} = v_1 v_2 \cdots v_n$ をカテゴリの列 $\mathbf{c} = c_1 c_2 \cdots c_n \ \ (c_i \in C)$ に変換する問題を考える。一般に、特徴ベクトル空間は量子化される。量子化の結果得られる部分特徴空間の集合を S とするとフレーズの観測値は、部分空間の列 $\mathbf{s} = s_1 s_2 \cdots s_n \ \ (s_i \in S)$ となる。このようなオブジェクト列の分類は、音声認識や各種のパターン認識問題に見られるように、以下に示される最適化問題として扱われることが多。

$$\arg\max_{\mathbf{c}} P(\mathbf{c})P(\mathbf{s}|\mathbf{c})$$

しかし、ここでは、フレーズ系列としての分類よりも、個々のフレーズの分類が目的となっているため、以下のような問題として定式化する。つまり、部分特徴空間の列 $\mathbf{s} \equiv s_1 s_2 \cdots s_n$ が与えられたとき、各 s_i に対して、以下の最適なカテゴリ c_i^* を求める問題と定義する。

$$\begin{aligned} c_i^* &\equiv \arg\max_{c \in C} \sum_{\{\mathbf{c} \mid \mathbf{c}[i]=c\}} P(\mathbf{c}|\mathbf{s}) \\ &= \arg\max_{c \in C} \sum_{\{\mathbf{c} \mid \mathbf{c}[i]=c\}} \frac{P(\mathbf{s}|\mathbf{c})P(\mathbf{c})}{P(\mathbf{s})} \\ &= \arg\max_{c \in C} \sum_{\{\mathbf{c} \mid \mathbf{c}[i]=c\}} P(\mathbf{s}|\mathbf{c})P(\mathbf{c}) \quad (1) \end{aligned}$$

ここで、$\mathbf{c}$ は長さ n のカテゴリ列を表し、$\mathbf{c}[i]$ はその第 i 成分を表している。この問題を解くためには、

1. 特徴空間の量子化
2. 条件付き確率 $P(\mathbf{s}|\mathbf{c})$ の推定法
3. 生起確率 $P(\mathbf{c})$ の推定法

が必要になる。なお、情報検索においては、特にサビの抽出が重要になるため、本稿では、フレーズをサビとそれ以外のフレーズに分類する 2 分類問題を考える。

特徴空間の量子化と条件付き確率を推定には、一般の分類手法を適用することができる。さまざまな分類法が存在するが、現在のところ我々のシステムでは、決定木を用いている。まず、切り出されたフレーズを以下の 4 つの特徴量を用いてベクトル化する。

- フレーズ内の相対的な平均音高値 (pitch)：フレーズ内の音符の音高値の平均値と楽曲全体の平均値の差を標準偏差で正規化した値
- フレーズ内の相対的な平均音長値 (duration)：フレーズ内の音符の音長値の平均値と楽曲全体の平均値の差を標準偏差で正規化した値
- フレーズ内の相対的な平均音強値 (velocity)：フレーズ内の音符の音強値の平均値と楽曲全体の平均値の差を標準偏差で正規化した値
- クラスタサイズ (frequency): フレーズに対応する prefix cluster の大きさ

フレーズを上記の 4 次元の特徴ベクトルに変換した後、サビとそれ以外のフレーズに分類するための決定木を訓練データから構築する。図 5 に学習

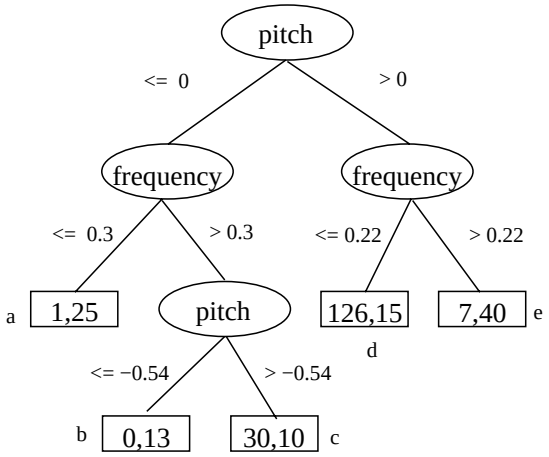

図 5: 特徴空間量子化のための決定木の例

の結果得られた決定木を単純化したものを示す。決定木の各葉は、特徴空間の部分空間に対応しており、この葉の集合を部分特徴空間の集合 S として用いる。

条件付確率 $P(\mathbf{s}|\mathbf{c})$ については、

$$P(s_1|c_1)P(s_2|c_2)\cdots P(s_n|c_n) \qquad (2)$$

で近似し、各カテゴリ c と決定木の葉 s に対して以下の式を用いて条件付き確率 $P(s|c)$ を訓練データから推定する。

$$\frac{f_{c,s}}{f_c}$$

ここで、f_c は、訓練データ中に含まれていたカテゴリ c のフレーズの数を、また、f_c は、訓練データ中に含まれていたカテゴリ c のフレーズで葉 c の条件にマッチする特徴ベクトルを持っているものの数を表している。

一方、生起確率の推定には、HMM を用いる。図 6 に、サビを抽出するための HMM を示す。この HMM では、サビを「+」、それ以外のフレーズを「-」で表している。状態 q_1 と q_2 はサビ以外のフレーズに対応する状態で、q_1 はメロディの先頭フレーズを、q_2 はそれ以外のフレーズを表している。同様に、状態 q_3 と q_4 はサビに対応する状態で、q_4 はメロディの先頭フレーズを、q_3 はそれ以外のフレーズを表している。

式 (1) の意味で最適な分類を行うためのアルゴリズムは、HMM の学習に用いられる Baum-Welch アルゴリズム [1] を若干修正することによって効率的に求めることができる。まず、フォワード確率

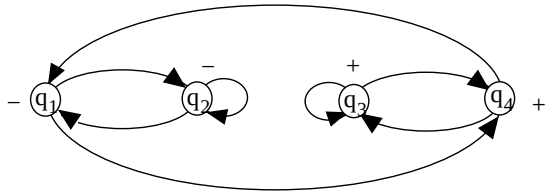

図 6: サビ抽出のための HMM

を F_q^k を図 6 において、任意の初期状態から k 回の遷移の後に状態 q に到達し、その出力記号列が観測された部分空間列 $\mathbf{s}$ の長さ k の接頭列 $\mathbf{s}_1^k$ に対応している確率とする。形式的には、以下の式で表される。

$$\sum_{\mathbf{c}\in C^k}\sum_{\mathbf{t}\in T_q^k} P(\mathbf{s}_1^k|\mathbf{c})P(\mathbf{c},\mathbf{t})$$

ここで、T_q^k は、任意の初期状態から k 回の遷移の後に状態 q に到達するすべての状態遷移系列の集合を表している。また、$P(\mathbf{c},\mathbf{t})$ は状態遷移系列 $\mathbf{t}$ において出力カテゴリ系列が $\mathbf{c}$ である同時確率を表している。式 (2) の近似のもとで、このフォワード確率は、以下の再帰的な形式で表すことができる。

$$\begin{aligned} F_q^0 &= t(0,q) \\ F_q^k &= \sum_{r\in Q} t(r,q)F_r^{k-1}\sum_{c\in C} o(r,c)P(s_k|c) \end{aligned}$$

ここで、$t(r,q)$ は HMM の状態 r から q への遷移確率を表し、$r=0$ のときは状態 q の初期確率を表している。一方 $o(q,c)$ は、状態 q における c の出力確率を表しており、$P(s_k|c)$ は、カテゴリ c のフレーズが決定木の葉 s_k に属する確率を表している。また Q は状態の集合を表している。

同様に、バックワード確率 B_q^k を図 6 において、状態 q 回から $n-k$ 回の状態遷移で部分空間列 $\mathbf{s}$ の長さ $n-k$ の接尾列 $\mathbf{s}_{k+1}^n$ に対応する出力記号列を出力する確率とする。するとバックワード確率も、以下の再帰的な形式で表すことができる。

$$\begin{aligned} B_q^n &= 1 \\ B_q^k &= \sum_{r\in Q} t(q,r)B_r^{k+1}\sum_{c\in C} o(q,c)P(s_{k+1}|c) \end{aligned}$$

フォワードおよびバックワード確率を用いて式 (1) の確率は以下のように表すことができる。

$$\sum_{\{\mathbf{c}\mid \mathbf{c}[k]=c\}} P(\mathbf{s}|\mathbf{c})P(\mathbf{c})$$

pitch	velocity	duration	frequency
0.828	0.730	0.696	0.870

表 1: Experimental Results

$$= \sum_{q \in Q} \sum_{r \in Q} F_q^{k-1} t(q,r) P(x_k|a) o(q,a) B_r^k \quad (3)$$

与えられた部分空間 s_i について、各カテゴリ c に対する確率 (3) を計算し、その値が最大となるカテゴリを s_i に対応するカテゴリとすることによって (1) の分類問題を解くことができる。

4 実験結果

本論文で述べたフレーズ分類法を日本のポピュラーミュージック 94 曲に適用し、フレーズ分類の実験を行った。実験に用いた楽曲は全体で 886 のフレーズを含んでおり、各フレーズに正解データを付与し、分類精度を評価した。実験では、94 曲をランダムに５つのグループに分け、クロスバリデーション法によって、精度の測定を行った。まず、特徴ベクトルに用いた各特徴のクロスバリデーションによる平均分類精度を表 1 に示す。表に示されるように特徴によって分類精度は異なるが、音高と prefix cluster のサイズが比較的有効な特徴量となっている。

次に、決定木を分類規則として用い各フレーズを独立に分類した場合の分類精度と HMM によって表されるフレーズ列の特性も加味した提案手法による分類精度を比較した。決定木のみを用いた場合の分類精度は平均 88.4% であったのに対して、フレーズ列の特性を加味した場合は平均 95.2%となった。

5 おわり

本稿では、音楽検索において問い合わせとデータベースの楽曲とをフレーズ単位でマッチングする方法を述べ、問い合わせとして与えられる可能性が高いサビを抽出する方法を述べた。現在のところフレーズがサビであることを用いことによって検索性能がどの程度向上するかを示す実験的な結果は出ていないが、高精度でサビフレーズを抽出できることを実験的に示した。

本稿の段階では、非常に小規模なデータに関する実験しか行われておらず、提案手法の有効性を示すには、より規模の大きいデータに対して実験することが必要である。現在、実験に用いる楽曲の収集を行っており、もう少し規模の大きなデータに対して再度実験を行うことを計画している。また、音楽検索の結果のランキングにフレーズのカテゴリを使うことの有効性の評価についても併せて実験することを計画している。

フレーズ分類の技術的側面に関しては、特徴空間の量子化および条件付き確率を求めるための決定木には、通常の分類問題で用いられる枝刈りと同じ手法を用いている。シークエンスデータの解析に適した決定木の枝刈り法の検討は興味深い問題と考えている。

参考文献

[1] L. E. Baum. An Inequality and Associated Maximization Technique in Statistical Estimation for Probabilistic Functions of a markov Process. *Inequalities*, 3:1 – 8, 1972.

[2] A. Ghias, J. Logan, D. Chamberlin, and B. C. Smith. Query By Humming – Musical Information Retrieval in an Audio Database. In *Proc. of ACM Multimedia'95*, 1995.

[3] Karen Kukich. "Techniques for Automtically Correcting Words in Text". *ACM Computing Surveys*, 24(4):377–439, 1992.

[4] T. Yanase, A. Takasu, and J. Adachi. Phrase Based Feature Extraction for Musical Information Retrieval. In *IEEE Pacific Rim Conference on Communications, Computers and Signal Processing (PACRIM'99)*, pages 396–399, 1999.

[5] 辻康博, 星守, 大森 匡. 曲の局所パターン特徴量を用いた類似曲検索・感性語による検索. **信学技法**, SP96-124:17–24, 1997.

[6] 柳瀬 隆史. **演奏情報からの特徴抽出によるメロディ検索システム**. 東京大学大学院工学系研究科修士論文, 1999.