

# 楽曲進行の推定による複旋律分類

吉原 幸輝<sup>†</sup> 三浦 孝夫<sup>†</sup> 塩谷 勇<sup>††</sup>

<sup>†</sup> 法政大学 工学研究科 電気工学専攻 〒184-8584 東京都小金井市梶野町 3-7-2

<sup>††</sup> 産業能率大学 経営情報学部 〒259-1197 神奈川県伊勢原市上粕屋 1573

E-mail: <sup>†</sup>{i05r3247,miurat}@k.hosei.ac.jp, <sup>††</sup>shioya@mi.sanno.ac.jp

あらまし 本稿では、複旋律を分類することを提案する。主なアイデアは、各小節から音集合の特徴量を抽出し、特徴量間の類似性を考慮しつつ、隠れマルコフモデルに基づいてモデルを生成し、分類基準を設定することにある。提案手法は、小節の長さごとに実験で検証され、高い正解率を得ることを示す。

キーワード 旋律分類, 旋律特徴量, マルコフ過程, 隠れマルコフモデル分類

## Classifying Polyphonic Melodies by Estimate of Music Process

Yukiteru YOSHIHARA<sup>†</sup>, Takao MIURA<sup>†</sup>, and Isamu SHIOYA<sup>††</sup>

<sup>†</sup> Dept. of Elect. & Elect. Engr., HOSEI University 3-7-2, KajinoCho, Koganei, Tokyo, 184-8584 Japan

<sup>††</sup> Department of Management and Information Science, SANNO University 1573, Kamikasuya, Isehara city, Kanagawa, 259-1197 Japan

E-mail: <sup>†</sup>{i05r3247,miurat}@k.hosei.ac.jp, <sup>††</sup>shioya@mi.sanno.ac.jp

**Abstract** In this investigation we propose a new and novel approach for classifying polyphonic melodies. Our main idea comes from *Probability Stochastic Processes* using Hidden Markov models where the characteristic features of polyphonic melodies are extracted from each bar. The similarity among harmonies can be considered by means of the features. We show the effectiveness and the usefulness of the approach by some experimental results.

**Key words** Melody Classification, Melody Features, Markov process, Hidden Markov Modeling

### 1. 前書き

本稿では、複旋律で構成された楽曲から特徴値を抽出し、未知の複旋律を分類していずれかラベルを付与する手法（旋律の自動分類）を論じる。これらは、本来手書き文字認識や手書き絵による類似検索と同様に、内容検索を目的とし、曲名や作品番号といったメタ情報を仮定しない。

旋律は主要な印象を与える役割を有しており、この機能を用いた応用は広範囲に想定できる。例えば、音楽作業（作曲・編曲）に対して直接的な支援を行うことや、同じクラスの曲を検索し著作権に抵触しそうな候補の絞込みを行うことができる。実際、音楽における著作権管理では、歌詞と異なって、旋律で判断することが多い。また、インターネットを介して得た音楽コンテンツの自動管理においても旋律による手がかりを与えることができる。

本研究の目的は、複旋律楽曲において確率過程を適用するのに適した特徴量を抽出し、確率過程に基づき旋律を分類することにある。具体的には、複旋律楽曲において小節単位で特徴量を抽出し、隠れマルコフモデルに基づき、生成確率を計算することで旋律の分類を行う。また同時に、未知旋律に対してコー

ドの推定も行う。

本稿では、音楽に関する基本知識 [26]、および情報検索技術の基本知識を仮定する [5], [29]。

第 2 章で、準備と必要な定義を述べ、第 3 章で音楽情報の特徴量を要約し、本研究で扱う特徴量を定義する。第 4 章では、主として分類に利用するマルコフ過程および隠れマルコフモデルの適用方法を示し、この特徴や利点を述べる。第 5 章では実験結果を述べ、本手法の有効性を示す。

### 2. 旋律の表現

本稿における旋律 (melody) とは、全ての音に対して開始時間や高さ (pitch)、長さ (duration) を表すシンボリックなものである。単旋律 (monophony) とは、同時に発生する音はせいぜい 1 音しかないものを指す。ホモフォニー (homophony) とは、同時に発生する音はせいぜい 1 組の和音しかないものを指し、和音間の音の重なりはないものとする。複旋律 (polyphony) とは、これらの制限がないものとする。コード (Chord, 和音) とは、3 音以上の音の積み重ね、またその響きを指す。全てのコードには基準となる音があり、これをルート (Root) という。

楽曲を計算機で表現するには、これまで信号レベル (CD 等

でのアナログ的变化を表現)、操作レベル (MIDI など、輝度・時間のゆれを表現) 及び楽譜レベル (表現・解釈の多様性を表現) でなされている。本稿では、特徴量の記述を目的とするため、旋律を五線譜を用いた楽譜レベルの表現を利用する。

### 3. 旋律特徴量

旋律を分類するためには、当該音楽の有する意味表現を考慮する必要がある。しかし旋律を分類するために記述される特徴量としてどのようなものが有効なのであるか? 旋律を分類するために指定される条件は、不完全である場合が多く、その大半は部分的である。

本研究では、旋律から特徴量を抽出し、ベクタ空間モデル [5], [29] を用いて特徴項目ベクトル空間へ写像する。各次元は特徴項目に対応するため、どの旋律も空間内の点で表せる。旋律質問の解はベクトル空間内の類似した点に対するランク付けで得られる [20]。空間内では伝統的な余弦質問で定義する。即ち  $\cos \theta = \frac{X \cdot Y}{|X| |Y|}$  を計算し、1.0 に近いほど類似していると判断する。ここで  $X, Y$  は空間ベクトルを、 $X \cdot Y$  は内積を、 $|X|$  はベクトル長を表す。

#### 3.1 単旋律における特徴量

これまで提案された旋律のための特徴量記述では、旋律輪郭 (Melody Contour) が代表的である [1], [7], [18]。旋律の輪郭情報のうち、ピッチ輪郭 (pitch contour) 表現は多くの研究で検討されている。ピッチ輪郭表現とは、単旋律にだけ有効であり、直前の音と比べてピッチが相対的に”高い”, ”低い”, ”同じ” という状態をそれぞれ文字 U, D, S で表し、輪郭を文字列として記述する<sup>(注 1)</sup>。この表現は、ピッチに対して相対的なので移調に強いが、雑音に反応して文字列が変わりやすく、旋律検索は不完全文字列一致処理の実行を意味する。

しかし、ピッチ輪郭方式は数多くの問題を含む [17]。大規模楽曲の記述から主旋律を抽出することは難しいため、多声音楽から各声を旋律と見なした輪郭を記述し、これらの結合から主旋律を想定する、という組み合わせ的探索を行わざるを得ない [2], [3]。旋律記述の記述に拍子・音符長を含まないため、判別のため多くの情報を必要とする。また、不完全一致検索手続きは多項式時間でよいことが知られるものの複雑で、(全走査を除けば) 2 次記憶域上での処理効率を向上させる方法は知られていない。輪郭抽出には雑音除去のための”しきい値”の設定が容易ではなく、生成される輪郭を規定する。

旋律が五線譜による楽譜で記述されるとき、小節単位にピッチスペクトル (pitch spectrum) を導入し特徴量とすることができる [10]。ピッチスペクトルとは、小節内に生じる各音符を長さで集積したヒストグラムを言う。このアイデアは信号処理では提案されているが、楽譜レベルで特徴量化するというアプローチはない [22]。ピッチスペクトルは (ギターコードのような) 小節全体に和音を指定したものとは異なる。ピッチ輪郭とは異なって、小節単位の特徴を集約しているため、旋律の輪郭を表してはいない。むしろ、小節間の変化を記述しているため、

旋律の時系列表現とも異なる。ピッチスペクトルは不完全な旋律に生じる問題を解消することができる。実際、装飾音・旋律・長短調・揺らぎがピッチスペクトルで局所化する。

ピッチスペクトルの問題は 移調に変動しやすいこと、および長調・短調の変化に対応しづらいことであることが知られている [10]。前者については、旋律から主音 (キー) を決定しテーマへの移調を行えばよい。しかし後者の問題については対応が難しく、より高度な対応が必要である。

#### 3.2 複旋律における特徴量

複旋律は、単旋律やホモフォニーのような制限事項がないため、表現される旋律の構成は、無限に近い組み合わせが考えられる。音の構成要素が複雑に関係しているため、単旋律における手法を直接適用することができない。そこで、楽譜をあらかじめ処理することによって対応する方法が考えられる [14]。前処理を行うことで、確率的に計算することが可能となる。

本稿で提案する特徴量は、各小節内で出現する音すべてをベクトルに射影し、音長のスペクトル列で表現することを提案する。この手法による特徴量は、現れる全ての音から構成され、また小節内の音の構成も見ることができるといった利点がある。

[例 1] カエルの歌を複旋律情報で表現すると次のようになる。

{C:1, D:1, E:1, F:1}, {C:2, D:2, E:2, F:1},  
 {C:1, D:1, E:2, F:1, G:1, A:1}, {E:2, F:2,  
 G:2, A:1}, {C:2, E:1, F:1, G:1}, {C:4},  
 {C:3, D:1, E:1, F:1}, {C:2, D:2, E:2, F:1}



図 1 カエルの歌五線譜

#### 3.3 本稿で提案する旋律特徴量

本稿では、旋律分類を行うための特徴量を、各小節内で出現する音から構成する手法を導入し、マルコフモデルによって分類精度を向上できるかを論じる。

各小節内で出現する音すべてをベクトルに射影し、音長のスペクトル列で表現する方法を提案したが、さらに、音長上位  $n$  音にしぼり、その  $n$  音で構成された集合を特徴量 (小節構成音) と定義する。小節単位の音情報を利用することで、主旋律の進行や特徴を得られると考えられる。ここで構成された小節構成音とは、音楽理論とは無関係の新しい和音の要素である。また、小節構成音となり得る候補が複数ある場合は、音の高いものから構成される音の組み合わせを選択する。

[例 2] カエルの歌の先頭 4 小節を、小節内の出現音の長さ上位 3 音を小節構成音とすると次のようになる。

1 小節目: {C:1, D:1, E:1, F:1} = {DEF}

(注 1): 開始文字は相対位置を定めることができないので - で表す。

2 小節目:  $\{C: 2, D: 2, E: 2, F: 1\} = \{CDE\}$

3 小節目:  $\{E: 1, F: 1, G: 1, A: 1\} = \{FGA\}$

4 小節目:  $\{E: 1, F: 1, G: 1\} = \{EFG\}$

#### 4. 隠れマルコフモデル

以下では、あらかじめクラスが与えられている旋律  $d$  を用いて、未知旋律をいずれかのクラスに分類する。本研究では、隠れマルコフモデルに基づいた分類手法を用いる。 $d$  は、特徴量  $w_j$  を用いた多重集合  $\{w_1, \dots, w_m\}$  と仮定する。

##### 4.1 隠れマルコフモデル

隠れマルコフモデルとは、実際に観測された出力記号系列から、その出力記号系列が生成されるための状態遷移系列モデルを推測・生成する手順である。一般に、与えられた記号系列からモデルのパラメータを決定することになる。本稿では、状態遷移系列をあらかじめ付与し Forward アルゴリズム (前向きアルゴリズム)、Viterbi アルゴリズムを用いてパラメータを推定する。Forward アルゴリズムとは、以下のようなステップで確率を再帰的に計算するアルゴリズムである。

(1) 各状態  $i=1, \dots, N$  に対して前向き確率を初期化:

$$\alpha_1(i) = \pi_i b_i(o_1)$$

(2) 各時刻  $t=1, \dots, T-1$ , 各状態  $j=1, \dots, N$  について、前向き確率を再帰的に計算:  $\alpha_{t+1}(j) = [\sum_{i=1}^N \alpha_t(i) a_{ij}] b_j(o_{t+1})$

(3) すべての累積確率を合計:  $P(o_1^T | M) = \sum_{i=1}^N \alpha_T(i)$

また Viterbi アルゴリズムは、以下のようなステップで求めるアルゴリズムである。

(1) 各状態  $i=1, \dots, N$  に対して前向き確率を初期化:

$$\delta_1(i) = \pi_i b_i(o_1)$$

(2) 各時刻  $t=1, \dots, T-1$ , 各状態  $j=1, \dots, N$  について、再帰的に計算:  $\delta_{t+1}(j) = \max_i [\delta_t(i) a_{ij}] b_j(o_{t+1})$

(3) 最終的な累積確率を計算:  $\hat{P} = \max \delta_T(i)$

Forward アルゴリズムと Viterbi アルゴリズムはいずれも再帰的に確率を求める計算方法であるが、大きく異なる点は Viterbi アルゴリズムでは状態遷移に対して最大の確率値を用いていることにある。

##### 4.2 旋律分類への適用

本稿で扱う分類操作は、平均律コードを状態列として割り当てることでモデルの生成を行う。そして生成されたモデルを用いて、クラス所属確率を推定する。隠れマルコフモデルを分類操作に適用するアルゴリズムは以下ようになる。

(1) 入力: クラスあり旋律集合, 未知旋律集合

(2) あらかじめクラスの与えられている旋律集合から、状態列となる平均律コードを付与しモデルを推定・生成

(3) (2) で生成されたモデル (遷移確率) を用いて、未知旋律のクラス所属確率  $P(c_i | d)$  を推定する

(4) 出力: 各未知旋律のラベル  $c_k$  は次のように定める:

$$c_k = \text{ArgMax}_{c \in C} P(c | d)$$

本稿では、主題と変奏曲を比較する際、変奏曲内の小節構成音で主題に現れない小節構成音が存在する可能性が十分にある。

これまでの手法を適用した場合、得られる確率が 0 になる可能性が非常に高い。この問題を解決するために、通常修正のいずれかの種類が紹介される、しかし、確率値が積計算によってより小さい値になりがちであるため、我々が予想しない分類結果になるかもしれない。従って、容易に値の修正を決定することができない。かわりに本稿では、それぞれの 1 対の小節構成音の間に類似性の考えを導入し、確率値を調整する。

2 つの小節構成音  $w, w'$  という条件のもとで、我々は類似性  $\text{sim}(w, w')$  を余弦値と定義する:

$$\text{sim}(w, w') = \frac{w \cdot w'}{\|w\| \|w'\|} \quad (1)$$

そして、遷移確率  $P(w_i | w_{i-1})$  に  $w_{i-1}$  から  $w_i$  までに起こるすべての可能な小節構成音の上に類似性を掛け合わせた値の和集合を  $P'(w_i | w_{i-1})$  と定義する。

$$P'(w_i | w_{i-1}) = \sum_g P(w_i | g) \times \text{sim}(g, w_{i-1})$$

この値は、ひとつの状態遷移の確率値と同等である。また、重みを加えられた生成確率は次で与えられる。余弦値の計算は、特徴を取り出す全ての旋律に対し行う必要がある。そして、新しい基準は最尤推定のもとで調整される。

$$P'(c | d) = \prod_{i=1}^{|m|} P'_c(w_i | w_{i-1}) \quad (2)$$

ここで重み付けされた確率  $P'(c | d)$  は、Forward アルゴリズムの  $\alpha_T(i)$ 、Viterbi アルゴリズムの  $\delta_T(i)$  に相当する値である。確率  $P'(c | d)$  は通常の確率値とは異なる値となるが、最尤推定を応用した手法である。

[例 3] 小節構成音  $\{CDE\} = \{x_1 y_1 z_1\}$  と小節構成音  $\{CDF\} = \{x_2 y_2 z_2\}$  の類似度は、以下のように計算される。ただし、 $x_1 = y_1 = z_1 = x_2 = y_2 = z_2 = 1$  のベクトルとする。

$$\cos \theta = \frac{x_1 x_2 + y_1 y_2 + z_1 z_2}{\sqrt{(x_1^2 + y_1^2 + z_1^2) \times (x_2^2 + y_2^2 + z_2^2)}} = 0.66$$

[例 4] 特徴量表記による隠れマルコフモデルを用いたクラス分類例を示す。“カエルの歌 (A)”, “キラキラ星変奏曲 (B)” をラベルとし、その旋律が  $d_1, d_2$  として与えられている ( $L_1 = \{d_1, d_2\}$ ) として、“歓喜の歌” ( $d_3$ ) をクラス分類する ( $L_2 = \{d_3\}$ )。ここで利用する小節構成音は 3 音とする。



図 2 モーツァルト:キラキラ星変奏曲

ABC 表記による先頭 8 小節、もしくは 9 小節の特徴値はそれぞれ次のようになる。

$$d_1: \{CDEF, -\}, \{EDC, CDEF\}, \{EFGA, EDC\}, \{GFE, EFGA\}, \{CC, GFE\}, \{CC, CC\},$$



図3 ベートーベン:交響曲第9番

{C/2C/2D/2D/2E/2E/2F/2F/2, CC},  
 {EDC, C/2C/2D/2D/2E/2E/2F/2F/2}, {CDE}  
 $d_2$ : {CCGG, CCEC}, {AAGG, FCEC}, {FFEE, DBCA},  
 {DD3/4E/4C2, FGC}, {CCGG, CCEC}, {AAGG, FCEC},  
 {FFEE, DBCA}, {DD3/4E/4C2, FGC}  
 $d_3$ : {FFGA}, {AGFE}, {DDEF}, {F3/2E/2E2},  
 {FFGA}, {AGFE}, {DDEF}, {E3/2D/2D2}

小節単位で出現音長をまとめ、その小節における小節構成音を決定する。この小節構成音が観測系列である(表2)。

小節	$d_1$	$d_2$	$d_3$
1	{C:1, D:1, E:1, F:1}	{C:5, E:1, G:2}	{F:2, G:1, A:1}
2	{C:2, D:2, E:2, F:1}	{C:2, E:1, F:1, G:2, A:2}	{E:1, F:1, G:1, A:1}
3	{C:1, D:1, E:2, F:1, G:1, A:1}	{C:1, D:1, E:2, F:2, A:1, B:1}	{D:2, E:1, F:1}
4	{E:2, F:2, G:2, A:1}	{C:13/4, D:7/4, E:1/4, F:1, G:1}	{E:5/2, F:3/2}
5	{C:2, E:1, F:1, G:1}	{C:5, E:1, G:2}	{F:2, G:1, A:1}
6	{C:4}	{C:2, E:1, F:1, G:2, A:2}	{E:1, F:1, G:1, A:1}
7	{C:3, D:1, E:1, F:1}	{C:1, D:1, E:2, F:2, A:1, B:1}	{D:2, E:1, F:1}
8	{C:2, D:2, E:2, F:1}	{C:13/4, D:7/4, E:1/4, F:1, G:1}	{D:5/2, E:3/2}
9	{C:1, D:1, E:1}		

表1 出現音

小節	1	2	3	4	5	6	7	8	9
$d_1$	DEF	CDE	EGA	EFG	CFG	C	CEF	CDE	CDE
$d_2$	C	CEG	CFA	CEG	DFB	CEA	DFG	C	
$d_3$	FGA	FGA	DEF	EF	FGA	FGA	DEF	EF	

表2 小節構成音:観測系列

観測系列を用いて、ラベル付旋律について状態列となる平均律コード推定を行う。平均律コードの推定は、音楽理論に基づいて決定する(表3)。

小節	1	2	3	4	5	6	7	8	9
$d_1$	C	C	C	C	C	C	C	C	C
$d_2$	C	C	F	C	G	C	G	C	

表3 状態列:平均律コード

状態列が決定することで、それぞれの旋律において状態遷移図が生成できる(図4, 図5)。

生成された状態遷移図に基づいて、 $d_3$  にクラス割り当てを行う。 $d_3$  のクラス所属確率は、状態遷移図の積計算で求めるこ

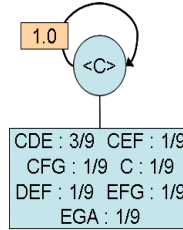


図4 カエルの歌:状態遷移図

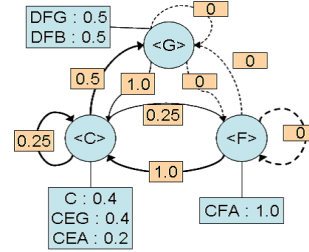


図5 キラキラ星変奏曲:状態遷移図

とが可能である。この時、各状態に所属する小節構成音も計算に用いられるが、状態の遷移が発生する際に小節構成音間の類似度も考慮することに注意を払う。以上の方法で  $d_3$  のクラス所属確率を求めると、それぞれ次の値となる。

前向きアルゴリズム (Forward Algorithm):

$$P(A|d_3) = 4.006 \times 10^{-3}$$

$$P(B|d_3) = 2.216 \times 10^{-3}$$

ビタビ・アルゴリズム (Viterbi Algorithm):

$$P(A|d_3) = 4.006 \times 10^{-3}$$

$$P(B|d_3) = 1.089 \times 10^{-3}$$

この結果、いずれの場合も  $d_3$  が最も近いクラスは A であり、“歓喜の歌” は “カエルの歌” に割り当てられた。

## 5. 実験

以下の実験では幾つかの変奏曲を扱う。変奏曲は1つのテーマと複数の変奏曲から構成されるため、ラベル付き訓練データはテーマの数だけあると考える。これ以外はラベルなしデータとみなし、旋律分類の対象とする。

### 5.1 準備

本稿では、複旋律で構成されたピアノ変奏曲を用いる。使用する楽曲は、Mozart キラキラ星変奏曲 K.265, Schubert 即興曲 Op.142-3, および Beethoven トルコ行進曲による6つの変奏曲 Op.76 の3つの変奏曲であり、それぞれ主題(テーマ)と12曲, 5曲, 6曲の変奏(パリエーション)からなる。これらは MIDI データとして表現されているものを ABC フォーマット [21] に変換して実験に使用する。

最初に、各楽曲をラベル付旋律集合とラベルなし旋律集合に分ける。ラベル付旋律集合は変奏曲テーマ(主旋律)3曲からなるとし、残り23の変奏曲はラベルなし旋律集合を構成する。各楽曲を表す ABC ファイルからラベル付旋律集合は全小節分

の特徴量，ラベルなし旋律集合は先頭 4 小節分の特徴量，または先頭 8 小節分の特徴量を取り出す．

本実験で抽出する特徴量は，小節内の出現音の長さ上位 3, 4, 5 音を，その小節の小節構成音とする．

また，ラベル付旋律集合の特徴量に対して，それぞれの状態を示す平均律コードをあらかじめ設定する．ここで得られた状態を用いて，各ラベルなし旋律集合について隠れマルコフモデルを用いて分類，あるいはコード進行の付与をする．

### 5.2 評価方法

本実験では 3 つのクラス "Mozart", "Schubert", "Beethoven" を設定し，隠れマルコフモデルを用いた分類器は全 23 変奏曲をいずれかのクラスに分類する．変奏曲が "正しく分類" できたとはそのテーマがラベルの作曲者によって作曲されたときとする．形式的に分類正解率を次のように定義する:

$$\frac{\text{正しく分類された変奏曲数}}{\text{総変奏曲数}}$$

本稿における総変奏曲数は 23 となる．また，本研究では，HMM 状態を平均律コードと対応させるため，変奏曲の各小節に対してコードが付与される．これは，従来"コード進行"分析(アナリゼ)と呼ばれる問題に対する自動解の生成に相当する．本研究で得られるコード進行と，実際のコード進行との一致率を小節単位で評価する．本実験では，8 小節の特徴量に付与されたコードを用い，分類結果で正解と判定された変奏曲を対象とする．

### 5.3 実験結果

状態列に平均律コードを用いた場合の先頭 4 小節を用いた際と先頭 8 小節を用いた際の生成確率を表 4 から表 7，図 6 から図 17 に，そして，各テーマに対する正解率を表 8 に，Viterbi アルゴリズムで付与されたコードの正解率を表 9 に示す．図の横軸は生成確率を表し，縦軸は分類する変奏曲番号を表す．

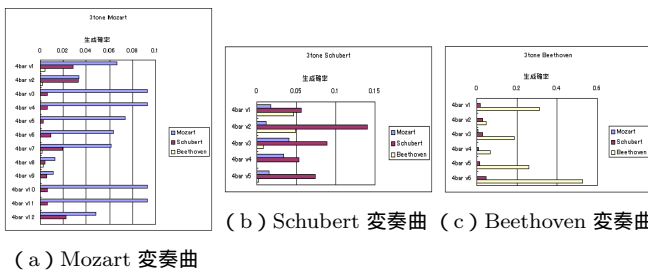


図 6 生成確率: 3 音, 4 小節, Forward

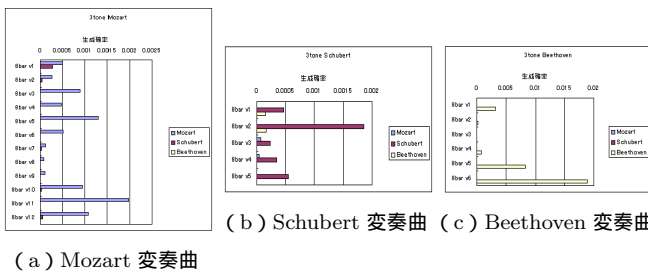


図 7 生成確率: 3 音, 8 小節, Forward

	Mozart	Schubert	Beethoven
ozart v1	$6.627 \times 10^{-2}$	$2.882 \times 10^{-2}$	$3.901 \times 10^{-3}$
v2	$3.389 \times 10^{-2}$	$3.316 \times 10^{-2}$	$2.231 \times 10^{-3}$
v3	$9.266 \times 10^{-2}$	$6.089 \times 10^{-3}$	$2.797 \times 10^{-4}$
v4	$9.266 \times 10^{-2}$	$6.089 \times 10^{-3}$	$2.797 \times 10^{-4}$
v5	$7.355 \times 10^{-2}$	$2.799 \times 10^{-3}$	$2.474 \times 10^{-4}$
v6	$6.311 \times 10^{-2}$	$9.422 \times 10^{-3}$	$2.253 \times 10^{-4}$
v7	$6.133 \times 10^{-2}$	$1.950 \times 10^{-2}$	$1.312 \times 10^{-3}$
v8	$1.292 \times 10^{-2}$	$4.068 \times 10^{-3}$	$2.559 \times 10^{-3}$
v9	$1.117 \times 10^{-2}$	$5.595 \times 10^{-3}$	$7.388 \times 10^{-4}$
v10	$9.266 \times 10^{-2}$	$6.089 \times 10^{-3}$	$2.797 \times 10^{-4}$
v11	$9.266 \times 10^{-2}$	$6.089 \times 10^{-3}$	$2.797 \times 10^{-4}$
v12	$4.840 \times 10^{-2}$	$2.222 \times 10^{-2}$	$7.638 \times 10^{-4}$
Schubert v1	$1.769 \times 10^{-2}$	$5.584 \times 10^{-2}$	$4.634 \times 10^{-2}$
v2	$1.199 \times 10^{-2}$	$1.403 \times 10^{-1}$	$4.859 \times 10^{-2}$
v3	$4.036 \times 10^{-2}$	$8.897 \times 10^{-2}$	$8.396 \times 10^{-3}$
v4	$3.391 \times 10^{-2}$	$5.263 \times 10^{-2}$	$6.980 \times 10^{-4}$
v5	$1.549 \times 10^{-2}$	$7.399 \times 10^{-2}$	$2.529 \times 10^{-3}$
Beethoven v1	$8.919 \times 10^{-4}$	$1.773 \times 10^{-2}$	$3.1070 \times 10^{-1}$
v2	$3.211 \times 10^{-3}$	$2.902 \times 10^{-2}$	$4.774 \times 10^{-2}$
v3	$5.077 \times 10^{-3}$	$3.037 \times 10^{-2}$	$1.872 \times 10^{-1}$
v4	$1.508 \times 10^{-3}$	$7.753 \times 10^{-3}$	$6.684 \times 10^{-2}$
v5	$1.522 \times 10^{-3}$	$1.473 \times 10^{-2}$	$2.592 \times 10^{-1}$
v6	$3.686 \times 10^{-4}$	$4.752 \times 10^{-2}$	$5.285 \times 10^{-1}$

表 4 生成確率: 3 音, 4 小節, Forward

	Mozart	Schubert	Beethoven
Mozart v1	$5.048 \times 10^{-4}$	$2.766 \times 10^{-4}$	$6.936 \times 10^{-7}$
v2	$2.655 \times 10^{-4}$	$4.119 \times 10^{-5}$	$1.325 \times 10^{-7}$
v3	$8.930 \times 10^{-4}$	$7.280 \times 10^{-6}$	$7.763 \times 10^{-8}$
v4	$4.683 \times 10^{-4}$	$7.563 \times 10^{-7}$	$2.109 \times 10^{-7}$
v5	$1.299 \times 10^{-3}$	$5.055 \times 10^{-6}$	$7.259 \times 10^{-8}$
v6	$5.186 \times 10^{-4}$	$1.010 \times 10^{-5}$	$2.453 \times 10^{-7}$
v7	$1.168 \times 10^{-4}$	$2.825 \times 10^{-5}$	$2.613 \times 10^{-6}$
v8	$7.936 \times 10^{-5}$	$2.646 \times 10^{-6}$	$3.523 \times 10^{-8}$
v9	$9.991 \times 10^{-5}$	$3.015 \times 10^{-6}$	$1.412 \times 10^{-7}$
v10	$9.513 \times 10^{-4}$	$2.377 \times 10^{-5}$	$9.096 \times 10^{-7}$
v11	$1.978 \times 10^{-3}$	$1.084 \times 10^{-5}$	$2.485 \times 10^{-9}$
v12	$1.078 \times 10^{-3}$	$4.747 \times 10^{-5}$	$8.706 \times 10^{-7}$
Schubert v1	$5.378 \times 10^{-6}$	$4.738 \times 10^{-4}$	$1.519 \times 10^{-4}$
v2	$2.884 \times 10^{-6}$	$1.854 \times 10^{-3}$	$1.643 \times 10^{-4}$
v3	$7.262 \times 10^{-5}$	$2.381 \times 10^{-4}$	$5.609 \times 10^{-7}$
v4	$3.996 \times 10^{-5}$	$3.424 \times 10^{-4}$	$1.892 \times 10^{-7}$
v5	$4.087 \times 10^{-6}$	$5.485 \times 10^{-4}$	$8.445 \times 10^{-6}$
Beethoven v1	$6.442 \times 10^{-8}$	$1.453 \times 10^{-5}$	$3.165 \times 10^{-3}$
v2	$2.094 \times 10^{-7}$	$3.980 \times 10^{-5}$	$1.796 \times 10^{-4}$
v3	$4.006 \times 10^{-6}$	$3.261 \times 10^{-5}$	$7.940 \times 10^{-5}$
v4	$4.548 \times 10^{-8}$	$6.053 \times 10^{-6}$	$8.211 \times 10^{-4}$
v5	$1.042 \times 10^{-7}$	$1.320 \times 10^{-5}$	$8.217 \times 10^{-3}$
v6	$8.681 \times 10^{-9}$	$1.434 \times 10^{-4}$	$1.894 \times 10^{-2}$

表 5 生成確率: 3 音, 8 小節, Forward

テーマ判定に関しては，Viterbi アルゴリズムを用いた場合，3 音の特徴量で小節数にかかわらず 100.0 % の高い正解率を示した．コード判定に関しては，3 音の特徴量で，4 小節以上

	Mozart	Shubert	Beethoven
Mozart v1	$2.007 \times 10^{-2}$	$1.791 \times 10^{-2}$	$2.435 \times 10^{-3}$
v2	$9.635 \times 10^{-3}$	$1.257 \times 10^{-2}$	$1.124 \times 10^{-3}$
v3	$2.572 \times 10^{-2}$	$3.243 \times 10^{-3}$	$2.341 \times 10^{-4}$
v4	$2.572 \times 10^{-2}$	$3.243 \times 10^{-3}$	$2.341 \times 10^{-4}$
v5	$3.150 \times 10^{-2}$	$1.390 \times 10^{-3}$	$2.150 \times 10^{-4}$
v6	$3.536 \times 10^{-2}$	$4.918 \times 10^{-3}$	$1.714 \times 10^{-4}$
v7	$3.074 \times 10^{-2}$	$9.694 \times 10^{-3}$	$4.724 \times 10^{-4}$
v8	$5.281 \times 10^{-3}$	$1.656 \times 10^{-3}$	$1.623 \times 10^{-3}$
v9	$6.166 \times 10^{-3}$	$2.895 \times 10^{-3}$	$7.388 \times 10^{-4}$
v10	$2.572 \times 10^{-2}$	$3.243 \times 10^{-3}$	$2.341 \times 10^{-4}$
v11	$2.572 \times 10^{-2}$	$3.243 \times 10^{-3}$	$2.341 \times 10^{-4}$
v12	$1.421 \times 10^{-2}$	$1.129 \times 10^{-2}$	$3.142 \times 10^{-4}$
Shubert v1	$1.073 \times 10^{-2}$	$2.715 \times 10^{-2}$	$2.035 \times 10^{-2}$
v2	$6.953 \times 10^{-3}$	$7.175 \times 10^{-2}$	$2.815 \times 10^{-2}$
v3	$1.365 \times 10^{-2}$	$4.380 \times 10^{-2}$	$4.257 \times 10^{-3}$
v4	$9.786 \times 10^{-3}$	$2.132 \times 10^{-2}$	$4.296 \times 10^{-4}$
v5	$9.481 \times 10^{-3}$	$3.753 \times 10^{-2}$	$1.272 \times 10^{-3}$
Beethoven v1	$4.199 \times 10^{-4}$	$5.182 \times 10^{-3}$	$1.329 \times 10^{-1}$
v2	$2.084 \times 10^{-3}$	$9.048 \times 10^{-3}$	$2.086 \times 10^{-2}$
v3	$2.959 \times 10^{-3}$	$1.011 \times 10^{-2}$	$8.609 \times 10^{-2}$
v4	$1.214 \times 10^{-3}$	$4.548 \times 10^{-3}$	$2.834 \times 10^{-2}$
v5	$9.885 \times 10^{-4}$	$5.067 \times 10^{-3}$	$9.776 \times 10^{-2}$
v6	$1.839 \times 10^{-4}$	$1.436 \times 10^{-2}$	$2.693 \times 10^{-1}$

表 6 生成確率: 3 音, 4 小節, Viterbi

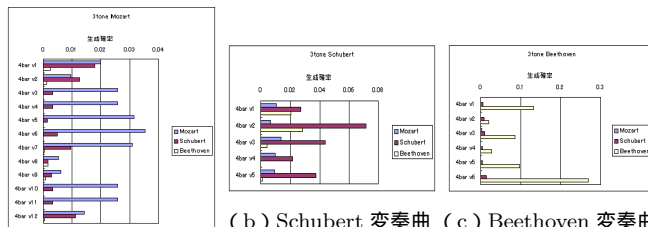
	Mozart	Shubert	Beethoven
Mozart v1	$1.381 \times 10^{-4}$	$9.486 \times 10^{-5}$	$1.883 \times 10^{-7}$
v2	$4.592 \times 10^{-5}$	$5.253 \times 10^{-6}$	$3.053 \times 10^{-8}$
v3	$2.029 \times 10^{-4}$	$1.739 \times 10^{-6}$	$3.411 \times 10^{-8}$
v4	$7.926 \times 10^{-5}$	$1.790 \times 10^{-7}$	$1.285 \times 10^{-7}$
v5	$5.239 \times 10^{-4}$	$1.205 \times 10^{-6}$	$2.489 \times 10^{-8}$
v6	$1.979 \times 10^{-4}$	$3.514 \times 10^{-6}$	$7.536 \times 10^{-8}$
v7	$3.293 \times 10^{-5}$	$3.905 \times 10^{-6}$	$3.714 \times 10^{-7}$
v8	$2.132 \times 10^{-5}$	$3.558 \times 10^{-7}$	$1.301 \times 10^{-8}$
v9	$3.726 \times 10^{-5}$	$4.417 \times 10^{-7}$	$8.013 \times 10^{-8}$
v10	$2.344 \times 10^{-4}$	$6.028 \times 10^{-6}$	$3.861 \times 10^{-7}$
v11	$4.790 \times 10^{-4}$	$2.549 \times 10^{-6}$	$1.708 \times 10^{-9}$
v12	$2.515 \times 10^{-4}$	$9.577 \times 10^{-6}$	$1.490 \times 10^{-7}$
Shubert v1	$2.955 \times 10^{-6}$	$7.940 \times 10^{-5}$	$5.691 \times 10^{-5}$
v2	$1.503 \times 10^{-6}$	$4.015 \times 10^{-4}$	$5.194 \times 10^{-5}$
v3	$2.746 \times 10^{-6}$	$4.152 \times 10^{-5}$	$1.165 \times 10^{-7}$
v4	$7.251 \times 10^{-6}$	$2.975 \times 10^{-5}$	$8.344 \times 10^{-8}$
v5	$2.226 \times 10^{-6}$	$9.481 \times 10^{-5}$	$3.326 \times 10^{-6}$
Beethoven v1	$1.150 \times 10^{-8}$	$9.574 \times 10^{-7}$	$5.460 \times 10^{-4}$
v2	$6.571 \times 10^{-8}$	$3.871 \times 10^{-6}$	$2.869 \times 10^{-5}$
v3	$1.863 \times 10^{-6}$	$2.741 \times 10^{-6}$	$1.424 \times 10^{-5}$
v4	$2.747 \times 10^{-8}$	$6.183 \times 10^{-7}$	$2.191 \times 10^{-4}$
v5	$3.606 \times 10^{-8}$	$1.869 \times 10^{-6}$	$1.064 \times 10^{-3}$
v6	$1.730 \times 10^{-9}$	$1.044 \times 10^{-5}$	$3.529 \times 10^{-3}$

表 7 生成確率: 3 音, 8 小節, Viterbi

の一致が 69.6 % の正解率となった。

#### 5.4 考察

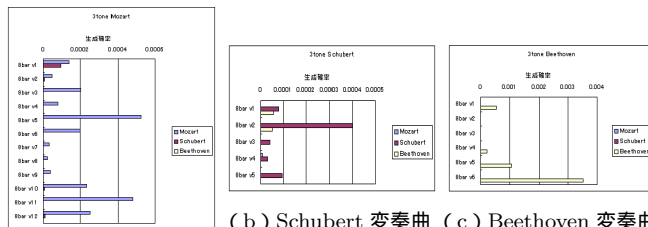
分類正解率の結果では、特徴量の小節数を増やすことで分



(a) Mozart 変奏曲

(b) Schubert 変奏曲 (c) Beethoven 変奏曲

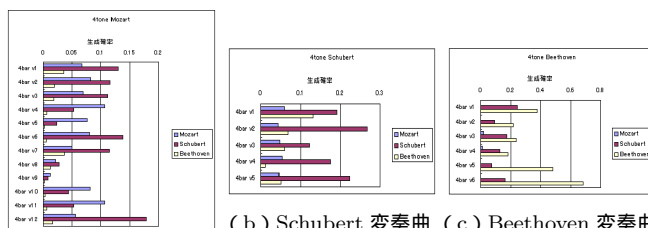
図 8 生成確率: 3 音, 4 小節, Viterbi



(a) Mozart 変奏曲

(b) Schubert 変奏曲 (c) Beethoven 変奏曲

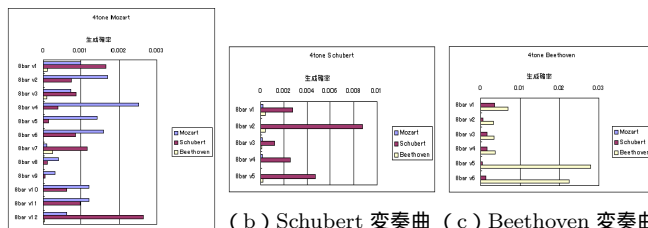
図 9 生成確率: 3 音, 8 小節, Viterbi



(a) Mozart 変奏曲

(b) Schubert 変奏曲 (c) Beethoven 変奏曲

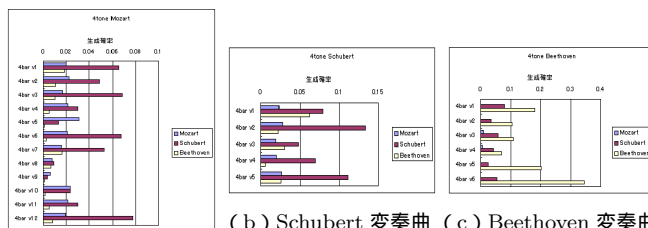
図 10 生成確率: 4 音, 4 小節, Forward



(a) Mozart 変奏曲

(b) Schubert 変奏曲 (c) Beethoven 変奏曲

図 11 生成確率: 4 音, 8 小節, Forward

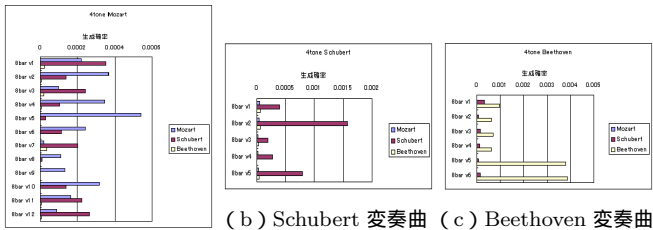


(a) Mozart 変奏曲

(b) Schubert 変奏曲 (c) Beethoven 変奏曲

図 12 生成確率: 4 音, 4 小節, Viterbi

類正解率の向上が見られた。具体的には、3 音の実験結果では、Viterbi アルゴリズムの場合で 95.7 % から 100.0 % に、4 音の実験結果では、Forward アルゴリズムの場合で 69.6 % から 82.6 % に、Viterbi アルゴリズムの場合で 56.5 % から 78.3 % に正解率が向上した。しかし 5 音の実験結果では、Forward アルゴリズムでは 39.1 % から 30.4 % と減少がみられた。この



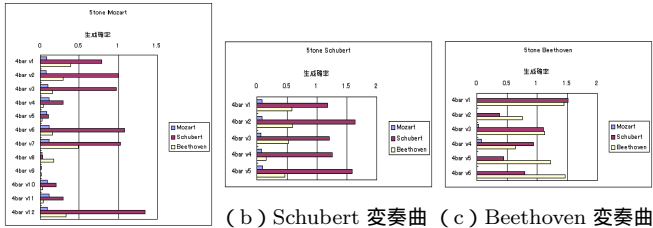
(a) Mozart 変奏曲

(b) Schubert 変奏曲 (c) Beethoven 変奏曲

図 13 生成確率: 4 音, 8 小節, Viterbi

() 内は小節構成音数	使用モデル	
	Chord(Forward)	(Viterbi)
(3) 4 小節	100.0 %	95.7 %
(3) 8 小節	100.0 %	100.0 %
(4) 4 小節	69.6 %	56.5 %
(4) 8 小節	82.6 %	78.3 %
(5) 4 小節	39.1 %	47.8 %
(5) 8 小節	30.4 %	47.8 %

表 8 正解率 / テーマ判定



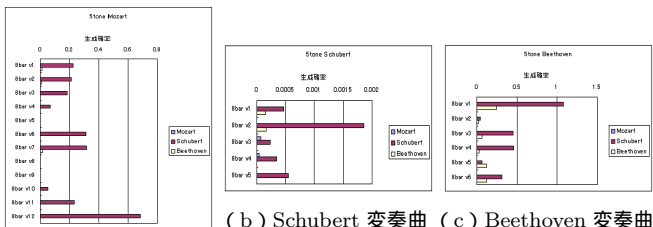
(a) Mozart 変奏曲

(b) Schubert 変奏曲 (c) Beethoven 変奏曲

図 14 生成確率: 5 音, 4 小節, Forward

	正解小節数		
	8 小節	6 小節以上	4 小節以上
3 音	4.35 %	39.1 %	69.6 %
4 音	0.0 %	38.9 %	55.6 %
5 音	0.0 %	45.5 %	63.6 %

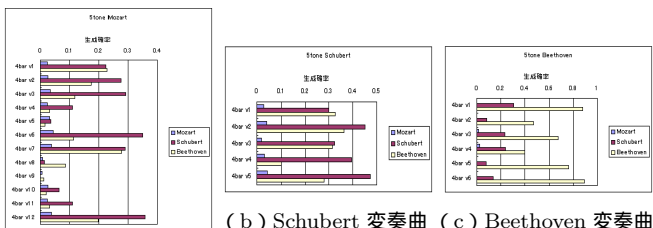
表 9 正解率 / コード判定



(a) Mozart 変奏曲

(b) Schubert 変奏曲 (c) Beethoven 変奏曲

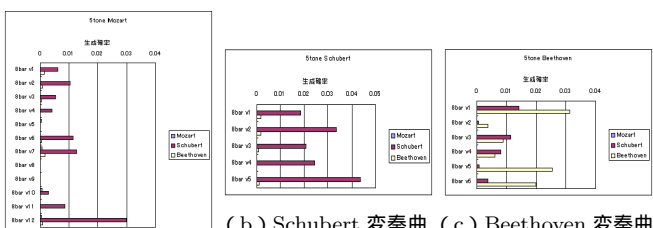
図 15 生成確率: 5 音, 8 小節, Forward



(a) Mozart 変奏曲

(b) Schubert 変奏曲 (c) Beethoven 変奏曲

図 16 生成確率: 5 音, 4 小節, Viterbi



(a) Mozart 変奏曲

(b) Schubert 変奏曲 (c) Beethoven 変奏曲

図 17 生成確率: 5 音, 8 小節, Viterbi

原因としていくつか考えられるが、小節構成音数が増加したことにより、他の変奏曲と大きく類似してしまう可能性がある。その結果として、小節数を増加するほど正解とは異なる変奏曲の生成確率が增大してしまい、誤判定をしたと考えられる。この問題は、以前の研究 [25] においても問題点としてあげている。

また、今回の実験では Forward アルゴリズムと Viterbi アル

ゴリズムの分類結果の比較も行った。特徴量が 3 音と 4 音に関しては、分類正解率の高さでは Forward アルゴリズムが高い値を示した。しかし、特徴量が 5 音に増加した場合には、Viterbi アルゴリズムが Forward アルゴリズムより高い分類率となった。また、Forward アルゴリズムでは小節数を増加させた結果、分類正解率の減少が起きたが、Viterbi アルゴリズムでは、向上まではいかないものの分類正解率を維持している。これは、アルゴリズムにおける計算手法の違いによるものと考えられる。Forward アルゴリズムで生成確率を計算する場合は、その値は全ての経路の総和になるのに対して、Viterbi アルゴリズムで生成確率を計算する場合は、最も確率の高い経路のみを選択することになる。今回扱った特徴量は出現音を組み合わせた形式であるが、その計算の中で互いの類似度を計算している。類似度計算で 0 確率を防ぐという利点がある一方、小節構成音数が増加していくにつれ互いに類似しあう音が増え、小節ごとにある特徴量が類似しやすく、あいまいさが増してしまう。さらに、Forward アルゴリズムのように総経路をたどった場合には、本来とは異なる小節構成音の確率が大幅に増大する可能性が考えられる。このことから、本研究で用いたような音を組み合わせた形式による特徴量を扱う上では、音数が少量で構成される場合は Forward アルゴリズム、多量の音数で構成される場合は Viterbi アルゴリズムが望ましいと考えられる。

正解率の比較として、単旋律曲の分類において最も正解率の高かった特徴量を先頭 4 小節でピッチスペクトラムを用い、分類手法に単純ベイズ、および EM アルゴリズムを用いた場合の正解率 [23] を表 10 に示す。

類似度 (%)	EM loop					
	0	5	10	15	20	25
30	87	91.3	91.3	91.3	91.3	91.3
50	78.3	91.3	91.3	87.0	87.0	87.0

表 10 正解率: 単旋律, ピッチスペクトル

単純ベイズ法 (EM loop 0 回) で最高 87 %, EM アルゴリズムで最高 91.3 % を示したが、本研究の手法においても、単旋律から複旋律へと複雑化しているにもかかわらず特徴量抽出

を8小節用いた場合の正解率が100.0%と、単旋律での分類と比較しても互角以上の正解率となった。本研究の手法は、複旋律分類において有効な抽出方法といえよう。

コード判定におけるコード一致率については、多くの変奏曲において完全一致がないものの、特徴量の半数にあたる4小節以上の一致率は55.6%から69.6%という7割近い変奏曲において一致が見られた。不一致を起こした原因として、変奏曲ごとに一定の箇所における不一致が多く見られた。具体的にはMozart変奏曲では、多くの場合先頭小節のコードに不一致が見られた。これにより、完全一致の割合が大きく減少しているものと考えられる。またSchubert変奏曲では、5小節から8小節間において正解となるコード進行の遷移がテーマに近く消滅していることで、異なるコード進行を選択した結果、最終的なコードの不一致が見られた。これにより、一致率の低下に繋がったと考えられる。しかし、多少の誤差を考慮したうえでコード割当は、計算により求められることが可能であり、有効な手段であるといえよう。

## 6. 結 論

本研究では、あらかじめクラスが与えられた少量の複旋律楽曲から多く存在する未知旋律を分類する方法を提案した。具体的には、複旋律情報の楽曲を小節単位の音情報から構成した特徴量も用いることと、隠れマルコフモデルを用いてモデル推定を行い、生成されたモデルで計算することにより、分類結果で高い正解率が得られた。この結果、複旋律情報においても高い分類精度を得ることを確認した。また、小節構成音と少量の複旋律楽曲にあらかじめコードを付与することで、多くの未知旋律に新たにコードを付与する方法も提案した。この結果、半数以上の小節において高い一致率を得ることを確認した。

## 文 献

- [1] Dowling W.J.: Scale and Contour – two components of a theory of memory for melodies, *Psychological Reviews* 85-4, 1978, pp.341-354.
- [2] Droettboom, M. et al.: An Approach Towards A Polyphonic Music Retrieval System, *Intn'l Symp. on Music Information Retrieval (ISMIR)*, 2001.
- [3] Droettboom, M. et al.: Expressive and Efficient Retrieval of Symbolic Musical Data, *Intn'l Symp. on Music Information Retrieval (ISMIR)*, 2002.
- [4] Ghais, A. et al.: Query By Humming, *ACM Multimedia Conf.*, 1995.
- [5] Grossman, D., O'Frieder: Information Retrieval – Algorithms and Heuristics, Kluwer Academic Press, 1998.
- [6] Kageyama, T. et al.: Melody Retrieval With Humming, *ACM Intn'l Computer Music Conf. (ICMC)*, 1993.
- [7] Kim, Y. et al.: Analysis of A Contour-based Representation for Melody, *Intn'l Symp. on Music Information Retrieval (ISMIR)*, 2000.
- [8] Lie, L. et al.: A New Approach to Query By Humming in Music Retrieval, *Intn'l Conf on Multimedia and Expo (ICME)*, 2001.
- [9] Meek, C. et al.: Thematic Extractor, *Intn'l Symp. on Music Information Retrieval (ISMIR)*, 2001.
- [10] Miura, T. and Shioya, I.: Similarities among Melodies for Music Information Retrieval, *ACM Conf. on Information and Knowledge Management(CIKM)*, 2003.
- [11] Mitchell, T.: Machine Learning, McGrawHill Companies, 1997.
- [12] Nigam, K. McCallum, A. Thrun, I. Mitchell, T.: Text Classification from Labeled and Unlabeled Documents using EM, Kluwer Academic Publishers, Boston. Manufactured in The Netherlands.
- [13] Pickens, J.: A Comparison of Language Modelling and Probabilistic Text Information Retrieval Approaches to Monophonic Music Retrieval, *Intn'l Symp. on Music Information Retrieval (ISMIR)*, 2000.
- [14] Pickens, J. and Crawford, T.: Harmonic Models for Polyphonic Music Retrieval, *ACM Conf. on Information and Knowledge Management(CIKM)*, 2002.
- [15] Ramakrishnan, R. and Gehrke, J.: Database Management Systems (2nd ed) 2000, McGrawHill
- [16] Sonoda, T. et al.: A WWW-based Melody Retrieval System, *Intn'l Computer Music Conf. (ICMC)*, 1998.
- [17] Uitdenbogerd, A.L. et al.: Manipulation of Music For Melody Matching, *ACM MultiMedia Conf.*, 1998.
- [18] Uitdenbogerd, A.L. et al.: Matching Techniques for Large Music Databases, *ACM Multimedia Conf.*, 1999.
- [19] Uitdenbogerd, A.L. et al.: MUSIC IR: Past, Present and Future, *Intn'l Symp. on Music Information Retrieval (ISMIR)*, 2000.
- [20] Uitdenbogerd, A.L. et al.: Music Ranking Techniques Evaluated, *Intn'l Symp. on Music Information Retrieval (ISMIR)*, 2000.
- [21] Walshaw, C.: abc Version 1.6, <http://www.gre.ac.uk/~c.walshaw/abc2mtex/abc.txt>
- [22] Yang, C.: Music Database Retrieval Based on Spectral Similarity, *Intn'l Symp. on Music Information Retrieval (ISMIR)*, 2001.
- [23] Yoshihara, Y. and Miura, T.: Melody Classification Using EM Algorithm. *Computer Software and Applications Conference(COMPSAC)*, pp. 204-210, 2005.
- [24] Yoshihara, Y. and Miura, T.: Classifying Polyphonic Music Based on Markov Model. *Intelligent Data Engineering and Automated Learning(IDEAL)*, pp. 697-706, 2006.
- [25] 吉原 幸輝, 三浦 孝夫.: 確率過程に基づく複旋律分類, 日本データベース学会 Letters Vol.5, No.1, pp. 137-140, 2006.
- [26] 石桁他.: 新装楽典 - 理論と実習, 音楽之友社, 2001.
- [27] 岩崎学.: 不完全データの統計解析, エコノミクス社, 2002.
- [28] 上嶋 宏, 三浦 孝夫, 塩谷 勇.: 同義語, 多義語の考慮による文書分類の精度向上, 電子情報通信学会誌 Vol. J87-D-I No.2, 2004.
- [29] 北研二他.: 情報検索のアルゴリズム, 共立出版, 2002.
- [30] 新納 浩幸, 佐々木 捻.: EM アルゴリズムの最適ループ回数の予測を用いた語義判別規則の教師なし学習, 情報処理学会論文誌 Vol.44, No.12, 2003.