社団法人 情報処理学会 研究報告
IPSJ SIG Technical Report

2005－MUS－61 (15)
2005／8／5

# ドラムパターン推定によるドラム音認識誤り補正手法

吉井和佳† 後藤真孝‡ 駒谷和範† 尾形哲也† 奥乃博†

† 京都大学大学院 情報学研究科 知能情報学専攻 ‡ 産業技術総合研究所

yoshii@kuis.kyoto-u.ac.jp m.goto@aist.go.jp
{komatani, ogata, okuno}@i.kyoto-u.ac.jp

本稿では，認識誤りを含むドラム音の発音時刻列からドラムパターンを推定し，認識誤り補正を行う手法について述べる．本稿におけるドラムパターンとは，バスドラム音およびスネアドラム音の発音時刻列のペアで構成される周期的な時間構造のことを指す．まず，我々が提案したドラム音認識手法を音楽音響信号に適用してドラム音の発音時刻列を得る．次に，発音時刻列を短時間フーリエ解析して求まる周期長に基づき，ドラムパターンを切り出す．ここで，同じドラムパターンは連続して反復されやすいという仮定をおき，各ドラムパターン区間における実際の発音時刻列を推定する．最後に，切り出されたドラムパターンと推定された発音時刻列との比較により，認識誤りの可能性が高い時刻を検出し，再検証を行う．ポピュラー音楽50曲を用いたドラム音認識実験で，補正手法により認識率が77.4%から80.7%に改善することを確認した．

# An Error Correction Method of Drum Sound Recognition by Estimating Drum Patterns

KAZUYOSHI YOSHII†, MASATAKA GOTO‡, KAZUNORI KOMATANI†,
TETSUYA OGATA† and HIROSHI G. OKUNO†

† Dept. of Intelligence Science and Technology, Graduate School of Informatics, Kyoto University
‡ National Institute of Advanced Industrial Science and Technology (AIST)

This paper describes a method that estimates drum patterns from onset-time sequences of drum sounds that may include recognition errors and corrects them by using the drum patterns. In this paper, drum patterns are defined as periodic temporal structures which are pairs of onset-time sequences of bass and snare drum sounds. First, we apply our drum sound recognition method to musical audio signals, and obtain onset-time sequences of drum sounds. Next, we calculate the period length of those sequences by applying short-time Fourier transform, and extract drum patterns from them. Under the assumption that the same drum patterns tend to be repeated, we estimate an actual onset-time sequences in duration of each drum pattern. Finally, by comparing each drum pattern with its corresponding estimated onset-time sequences, we detect time points where recognition errors may have been made, and verify those points. The experiments of drum sound recognition with 50 popular songs showed that our correction method improved the recognition accuracy from 77.4% to 80.7%.

## 1. はじめに

近年，音楽情報検索（Music Information Retrieval: MIR）に関する研究がさかんである[1]．MIRでは，作曲者名や楽曲名だけでなく，リズムやメロディーなどの音楽コンテンツに基づいて検索を行うことを目指している．今日，携帯型音楽プレイヤーやディジタル音楽配信の普及によって我々をとりまく楽曲数は爆発的に増加しているため，効率的に楽曲を検索できるMIRの実現は急務である．とくに，作曲者名や楽曲名が付与されていない楽曲を対象とするには計算機が自動的に音楽コンテンツを記述できなければならない．

人間の音楽知覚に則したMIR実現のためには，我々は音楽の三大要素（リズム，メロディー，ハーモニー）の総合的理解が不可欠であると考えている．第一段階として我々はリズムに着目し，ポピュラー音楽においてリズムと密接な関係があるドラム音の認識手法を提案した[2]．この手法により，市販CDレベルの音楽音響信号からドラム音の発音時刻列を得ることができるようになった．しかし，リズム理解のためには，より高次の時間的コンテンツの記述が必要であった．また，認識精度の向上も課題として残されていた．

我々は今回，ドラム音の発音時刻列から一つ高次の時間的コンテンツとしてドラムパターンに着目する．

ドラムパターンとは，ドラム音の発音時刻列における周期的な時間構造である．例えば，人間がポピュラー音楽（あるいはドラム演奏）を聞いて手拍子（ビート）を打つ（1, 2, 3, 4, 1, 2, 3, ...）ことができるのは，ビートの周期性を把握し，次拍を予測しているからである．このことから，リズム知覚には音楽の時間方向の構造化（周期性の知覚）が重要であることが分かる．

後藤はビートトラッキングの研究[3]☆で，階層的なビート予測のためにドラム音の発音時刻検出を試みた．これは，ドラム音の発音時刻列には階層的な周期性があることを示している．また，Gouyon ら[4]は，リズムとは音楽イベント列における周期構造であり，階層的に組織化できると指摘した．すなわち，リズム理解には，ドラム音発音時刻の周期構造と階層構造の解析，すなわちドラムパターン推定が不可欠である．

このように，ドラムパターンは重要な時間的コンテンツであるにもかかわらず，テンポや拍子の記述に関する従来研究[5]では扱われてこなかった．後藤ら以降のビートトラッキングの研究は，特徴量ベースの周期性に着目しているもの[6]がほとんどであった．

ドラムパターンの周期性や遷移の方法をモデル化することができれば，さらに高次の時間的コンテンツ記述に役立つ．一方，そのようなモデルは，低次の時間的コンテンツである発音時刻列中の認識誤りを補正することにも役立つ．本稿では，同じドラムパターンは連続して反復されやすいという単純なモデルを仮定し，認識誤りを補正する手法を考案したので報告する．このモデル化はベースラインとなるもので，今後さらに洗練する余地が残されている．

まず2章で，認識誤り補正機能を含んだドラム音認識システムの全体像を示し，考察を加えるとともに，我々が以前提案した認識手法の概略を説明する．次に3章で，提案する認識誤り補正手法のアルゴリズムについて説明する．4章で補正手法の有効性を検証した実験について報告し，最後に5章でまとめとする．

## 2. 認識誤り補正機能つきドラム音認識システム

本システムは，**ドラム音認識部**が音楽音響信号中のバスドラム音，スネアドラム音の発音時刻を個別に検出し，**認識誤り補正部**がドラムパターン推定に基づく認識誤り補正を行う．以下に説明する．

### 2.1 スペクトログラムテンプレートに基づくドラム音認識

ドラム音認識部は，音楽音響信号を入力とし，バスドラム音とスネアドラム音の発音時刻列を出力する．このとき，ドラム音のスペクトログラムをテンプレートとするテンプレート適応部とテンプレートマッチング部が連続して動作する．以下に各部の概略を説明する（詳細は文献[2]を参照のこと）．

☆ 後藤らはドラム音を含まない楽曲に対してもビートトラッキングを試みている．すなわち，リズム理解にはドラム音に着目するだけでは十分ではないことを記しておく．

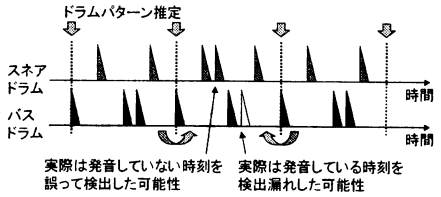


図1 ドラムパターン推定に基づく認識誤り補正

#### 2.1.1 テンプレート適応

テンプレートマッチングに用いるテンプレートとして，楽曲中のドラム音のスペクトログラムを得ることが目的である．事前に初期テンプレートとして，バスドラム音とスネアドラム音それぞれのスペクトログラムを1つずつ用意しておく（種テンプレートと呼ぶ）．現実には楽器の個体差のため，種テンプレートと楽曲中のドラム音のスペクトログラムは異なるので，種テンプレートに対する適応処理を行う．このとき，ドラム音の発音時刻を複数（全部ではない）検出することを試みるが，その精度が高いことが重要である．

#### 2.1.2 テンプレートマッチング

適応後テンプレートを時間方向にシフトしながら楽曲のパワースペクトルとの距離を計算し，距離がある閾値以下の時刻を発音時刻として出力する．楽曲のスペクトログラムはテンプレート以外の楽器音スペクトルを多く含んでいるので，周波数成分の重畳に対して頑健な距離尺度を利用する．また，距離の閾値は大津の自動閾値決定法[7]を利用して求める．

### 2.2 ドラムパターン推定に基づく認識誤り補正

本稿では，ドラム音の発音時刻列中の周期的な時間構造をドラムパターンと呼ぶ．これは，バスドラム音とスネアドラム音の発音時刻列のペアである．人間は同じドラムパターンは周期性を持つ（連続して反復される）と期待し，ドラム音の発音時刻予測に利用している．一方，このようなドラムパターンの性質はドラム音の発音時刻配置に関する制約であるといえる．すなわち，時間軸上にドラム音が発音しやすい時刻と発音しにくい時刻とが存在すると考え，これを推定してドラム音認識誤り補正の手がかりにする．

認識誤り補正部は，入力，出力ともにドラム音の発音時刻列である．図1に動作例を示す．まず，ドラム音認識部で得られた発音時刻列からドラムパターンを推定する（**ドラムパターン推定部**）．次に，推定されたドラムパターンの連続性を仮定して認識誤りの可能性が高い時刻を検出し，実際にドラム音が発音しているかの検証を行う（**発音検証部**）．

## 3. ドラムパターン推定に基づく認識誤り補正

認識誤り補正部は，ドラムパターン推定部と発音検証部からなる．以下に説明する．

### 3.1 実現上の課題

システム実現上の主な課題は以下の通りである．

**課題 1：ドラムパターンの開始時刻の定義と推定**　ドラムパターンの開始時刻として小節内 1 拍目を常に推定することは難しい．例えば，4/4 拍子の楽曲中で発音時刻列の周期長が 2 拍であるような区間では，小節内 3 拍目の時刻からドラムパターンが切り出される可能性を考慮する必要がある．

**課題 2：ドラムパターンの時間長の定義と推定**　ドラムパターンの時間長を小節長と定義するのは適切ではない．例えば，2/4 拍子の楽曲では，発音時刻列の周期長が 2 拍（小節長）であったり，4 拍（小節長の 2 倍）であったりする．2/4 拍子と 4/4 拍子の区別には，ドラム音の発音時刻より，拍の強中弱の配置が重要である楽曲が少なくない．

**課題 3：ドラムパターン遷移のモデル化**　直感的には，ポピュラー音楽では同じドラムパターンがよく繰り返されるように感じる．しかし実際には，繰り返されるドラムパターンは細部が少しずつ異なっていたり，周期は楽曲ごとに異なる．

**課題 4：認識誤りの可能性が高い時刻の検出**　検出されたが実際には発音していない可能性が高い時刻（suspicious onset と呼ぶ），検出されなかったが実際には発音している可能性が高い時刻（potential onset と呼ぶ）をそれぞれ検出する必要がある．

**課題 5：認識誤りの可能性が高い時刻の検証**　最初の発音時刻検出時（2.1 節）とは異なる判定基準（閾値など）を持つ検証方法が必要である．

### 3.2 解決方法

まず，入力音響信号に関して以下の制約を設ける．

**制約 1**　2/4 拍子か 4/4 拍子で，途中で変化しない．

**制約 2**　bpm は 60 から 200 の範囲である．

RWC 研究用音楽データベース（RWC-MDB-P-2001）[8] を調査したところ，上記制約はデータベース中ほぼすべてで成立することを確認した．市販のポピュラー音楽の大半でも当てはまると考えられる．また，ドラムパターンの性質として，同じドラムパターンが 4 拍単位で反復される可能性が高いことを仮定する．このような条件のもとで，図 2 に示すシステムを構築した．

#### 3.2.1 ドラムパターン推定部

ドラムパターン推定部では，**開始時刻は小節内 1 拍目，時間長は 4 拍分となるドラムパターン**を発音時刻列から切り出す（開始時刻と時間長を合わせて**属性**と呼ぶ）ことが目的である．あらかじめ，上記の属性の平均的なドラムパターン（**リファレンスパターン**と呼ぶ）を用意しておく．まず，発音時刻列に短時間フーリエ解析を行い，フレームごとに周期長を算出する．次に，リファレンスパターンの時間長を周期長の 2 の整数乗倍に伸長し，発音時刻列との相関値を逐一求める（時間長で正規化）．このとき，最大値をとる時間長（相関値が等しい場合，より長い時間長を選択）とそのときの相関値をフレームごとに記憶する．最後に，ある閾値以上の相関値をとるフレームを開始時刻とし，記憶した時間長をドラムパターンとして切り出す．

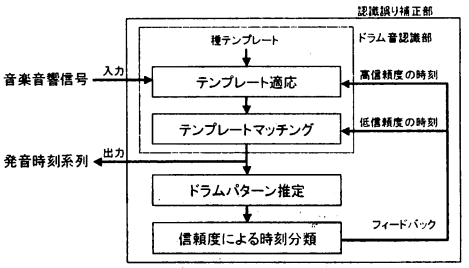


図 2　ドラムパターン推定に基づく認識誤り補正システム

これは，リファレンスパターンとの相関が高く，より時間長の長い発音時刻列を探索する処理である．例えば，周期長が 2 拍である区間でも，2 倍の 4 拍分のドラムパターンを切り出せる．すなわち，4/4 拍子なら 1 拍目から 4 拍分の，2/4 拍子なら 2 小節を連結した 4 拍分を切り出すことを試みる（課題 1,2 に対応）．

#### 3.2.2 発音検証部

発音検証部では，認識誤りの可能性が高い時刻の検出と検証を行う．まず，同一パターンが連続して反復されるという仮定をモデル化する（課題 3 に対応）．このモデルを用い，連続したパターン中の同じ位置に頻出する発音時刻を実際に発音している可能性が高い時刻として，そうでない時刻を認識誤りの可能性が高い時刻として検出する（課題 4 に対応）．もし，切り出されたドラムパターンが意図した属性でなくても（例：4/4 拍子区間で小節内 3 拍目から 4 拍分が切り出される），その前後に隣接する同じ属性のパターンを切り出せば，上記の検出処理は実行できる（課題 1,2 に対応）．

次に，発音している可能性が高い時刻は 2.1.1 節で述べた要求に合致するので，テンプレート適応部にフィードバックし，精度の高い適応後テンプレートを作り直す（課題 5 に対応）．ここで，このテンプレートを用いたテンプレートマッチングを行い，ベースとなる距離の閾値を自動的に求めておく．最後に，認識誤りの可能性が高い時刻に対してベース閾値を変化させ再度テンプレートマッチングをやり直す（課題 5 に対応）．すなわち，suspicious onset では閾値を下げて距離の大きい発音時刻を除去し，potential onset では閾値を上げて最初に検出されなかった発音時刻を検出する．

### 3.3 アルゴリズム

本システムにおける時間単位（1 フレーム）は 10 [ms] である．以下にドラムパターン推定アルゴリズムと発音検証アルゴリズムを示す．

#### 3.3.1 発音時刻分布の生成

バスドラム音，スネアドラム音の発音時刻列をそれぞれ $T_B, T_S$ とする．いま，演奏の揺らぎに起因する発音時刻誤差をガウス分布 $G$ でモデル化する．誤差モデル $G$ を発音時刻列中の各時刻に配置した時間列をそれぞれ $D_B, D_S$ とし，発音時刻分布と呼ぶ．本稿では，ガウス分布 $G$ の標準偏差は 2 [frames] とした．実際に発音時刻分布を生成した例を図 3 に示す．

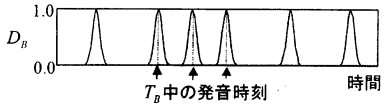


図 3　発音時刻分布列 $D_B$ の例

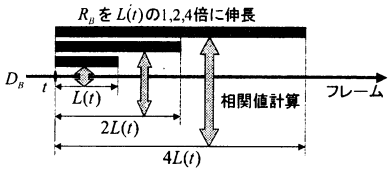


図 4　リファレンスパターン $R_B$ を 2 の整数乗倍に伸長しながら発音時刻分布 $D_B$ との相関値を計算

### 3.3.2 ドラムパターンの推定

ここでは，ドラムパターンの開始時刻（1 拍目）と時間長（4 拍分）を推定することが目的である．まず，発音時刻分布 $D_B, D_S$ に対し，ハニング窓，窓幅 2048 [frames]，窓シフト長 1 [frame] の STFT を適用して得られる振幅スペクトルを $A_B, A_D$，これらの和を $A$ とする．以下の手順 (1)，(2) を各フレームごとに行う．

(1) **周期長の計算**　まず，各フレーム $t$ における振幅スペクトル $A(t)$ の自己相関を計算し，周期長 $L(t)$ を求める．周期長列 $L$ の大部分は，周期長 $2^u L_p \pm \delta$（$L_p$ はある周期長，$u, \delta$ は非負整数で $\delta = 5$ とした）（例：$140 \pm \delta, 280 \pm \delta$ [frames]）となっていることに注意する．

(2) **リファレンスパターンとの相関値計算**　まず，4 拍分のリファレンスパターン $R_B, R_S$ を 120 [frames]（200 bpm に対応）から 400 [frames]（60 bpm に対応）の範囲で時間長 $2^v L(t)$ [frames]（$v$ は非負整数）（例：$L(t) = 142$ なら 142, 284 [frames]）に伸長させる（後述）．伸長後のリファレンスパターンとフレーム $t$ からの発音時刻分布 $D_B, D_S$ との相関値をそれぞれ計算し（図 4），これらの和を時間長で正規化したものを $C_v(t)$ とする．このとき，$C_v(t)$ が最大となる最大の $v$ を $v'(t)$ とし，そのときの相関値を $C(t)$ とする．

このようにして相関値列 $C$ が得られる．基本的には，相関値 $C(t)$ がある閾値以上のフレーム $t$ を開始時刻，$2^{v'(t)}L(t)$ を時間長としてドラムパターンを切り出していけばよい．ただし，切り出し区間が重なった場合，相関値が大きい方のドラムパターンを優先して切り出す．

### 3.3.3 リファレンスパターンの準備と伸長

リファレンスバスドラムパターン $R_B$，リファレンススネアドラムパターン $R_S$ は汎用性が持つことが望ましいので，大量のドラムパターンの平均で構成する．

まず，RWC 研究用音楽データベース（RWC-MDB-P-2001）[8] の MIDI ファイル中の 2/4 拍子と 4/4 拍子の区間を 4 拍長のセグメントに切り分ける（全 7819 セグメント）．次に，1 拍を 12 分割（4 拍で 48 分割）し，

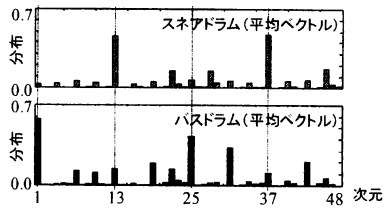


図 5　ドラム音の発音時刻分布（平均ベクトル）

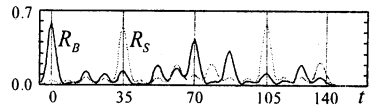


図 6　伸長後のリファレンスバスドラムパターン $R_B$ とリファレンススネアドラムパターン $R_S$ の例（時間長 140 [frames]）

各セグメントを 48 次元ベクトルに変換する．ベクトルの各次元の値は，その次元に対応する時間内でドラム音が発音していれば 1，発音していなければ 0 とする．これらの平均ベクトルを図 5 に示す．

時間長 $2^v L(t)$ のリファレンスパターンは，平均ベクトルを $2^v L(t)/48$ 倍に伸長し，ガウス分布 $G$ を畳み込みこんで生成する．リファレンスパターンの例（時間長 140 [frames]）を図 6 に示す．

### 3.3.4 実際の発音時刻分布の推定

あるドラムパターン区間の実際の発音時刻分布を推定するため，仮定に基づき，当該パターン周辺の発音時刻分布列の重み付き和モデルを提案する．

いま，$E_B, E_S$ をそれぞれバスドラム音，スネアドラム音の推定発音時刻分布と呼ぶ．これらは，切り出された各ドラムパターンに対応する区間の推定発音時刻分布を時間軸上に配置したものであり，次式で求める．

$$\begin{aligned} E(p_i + \lambda) = \quad & D(p_i - 2l_i + \lambda) * 0.25 \\ & + D(p_i - l_i + \lambda) * 0.25 \\ & + D(p_i + l_i + \lambda) * 0.25 \\ & + D(p_i + 2l_i + \lambda) * 0.25 \end{aligned} \tag{1}$$

ここで，$p_i$, $l_i$ $(i = 1, \cdots, N)$ とは切り出されたドラムパターンの開始時刻と時間長を示す（$N$ は個数）．また，添え字 $_{B,S}$ は省略して表記した（以降も同様）．

この推定方法を図 7 に図示する．推定対象の区間に隣接する区間だけでなく，さらに 1 つ隣の区間にも着目するのは，小節内で同じ拍となる区間を確実に含めるためである．例えば，当初の目的とは異なり，4/4 拍子の楽曲から 1,2 拍目がドラムパターンとして切り出された場合にも，そのパターンと同じ小節の 3,4 拍目だけでなく，次の小節の 1,2 拍目も推定に利用する．本稿では，4 区間の重みはすべて等しく 0.25 とした．

このモデルは，切り出される各ドラムパターンの属性が異なっていても適用可能である利点を持つ．今後，より高度なモデル化（確率的・統計的モデル化）を行うには，ドラムパターンの属性を統一する必要がある．

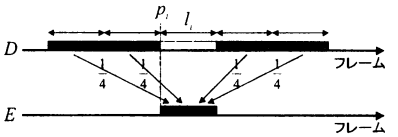


図 7 ドラムパターン $(p_i, l_i)$ 区間における発音時刻分布の推定：発音時刻分布 $D$ に基づく推定発音時刻分布 $E$ の計算

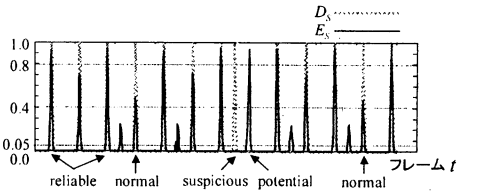


図 8 信頼度による発音時刻のクラス分けの例：発音時刻分布 $D_S$ 推定発音時刻分布 $E_S$ との比較

### 3.3.5 信頼度による時刻分類と発音検証

得られた推定発音時刻分布 $E_B, E_S$ と発音時刻分布 $D_B, D_S$ とをそれぞれ比較し，発音時刻列 $T_B, T_S$ を信頼度の点から以下のクラス (1)〜(3) に分類する．また，$T_B, T_S$ としては検出されなかった時刻として，新たにクラス (4) を考える．

(1) **reliable onset** 推定発音時刻分布に合致し，実際に発音している可能性が高い発音時刻．テンプレート適応部へフィードバックし，適応後テンプレートを再構成する．このテンプレートを用いて再度マッチングを行い，距離の閾値（$\Phi$ とする）を大津の方法[7]で自動的に求めておく．

(2) **normal onset** 中間的な信頼度の発音時刻．$\Phi$ を減少させてもう一度マッチングを行うことで，実際には発音していない時刻を除く．

(3) **suspicious onset** 推定発音時刻分布から逸脱し，実際に発音していない可能性が高い発音時刻．$\Phi$ をさらに減少させてもう一度マッチングを行う．

(4) **potential onset** 推定発音時刻分布に基づいて，実際には発音していると予測される潜在的な発音時刻．$\Phi$ を増加させてもう一度マッチングを行うことで，発音判定の許容度を上げる．

ここで，各クラスに属するバスドラム音の発音時刻を $T_B^{(1)}, T_B^{(2)}, T_B^{(3)}, T_B^{(4)}$ と表す．スネアドラム音に関しても同様に $T_S^{(1)}, \cdots, T_S^{(4)}$ と表す．本稿では，これらは以下のような閾値処理により求めた．

(1) $T^{(1)}$: $\{t|D(t) = 1.0,\ E(t) \geq 0.8\}$

(2) $T^{(2)}$: $\{t|D(t) = 1.0,\ 0.8 > E(t) \geq 0.05\}$

(3) $T^{(3)}$: $\{t|D(t) = 1.0,\ 0.05 > E(t)\}$

(4) $T^{(4)}$: $\{t|D(t) = 0.0,\ E(t) \geq 0.4\}$

発音時刻分布 $D_S$，推定発音時刻分布 $E_S$ を計算した例を図 8 に示す．RWC 研究用音楽データベース (RWC-MDB-P-2001) 50 曲中（表 1）の $T_B^{(1)}, \cdots, T_B^{(4)}$ に対する適合率は 91%, 77%, 66%, 23%，$T_S^{(1)}, \cdots, T_S^{(4)}$ に対する適合率は 92%, 37%, 51%, 6%であった．$T_B$ に対しては期待通りであったが，$T_S$ に対しては $T_S^{(2)}$ と $T_S^{(3)}$ が逆転したので，$T_S^{(2)}$ の閾値をより減少させることにした．この原因が認識誤りなのかスネアドラムパターンの性質なのかは，今後慎重な検討が必要である．

## 4. 評価実験

提案手法の有効性を評価するため，ポピュラー音楽音響信号を対象とした実験を行ったので報告する．

表 1 実験に用いた 50 曲 (RWC 研究用音楽データベース RWC-MDB-P-2001 より)

| piece number (No.) |
|---|
| 1,5,6,7,8,10,11,12,13,14,18,20,21,22,23,25,26,30, |
| 33,35,36,37,40,41,43,44,46,47,48,50,51,52,53,54, |
| 58,59,61,62,63,66,70,83,84,85,87,88,89,90,92,98 |

### 4.1 実験条件

入力音響信号として，後藤らの開発した RWC 研究用音楽データベース：ポピュラー音楽 (RWC-MDB-P-2001) [8] から 50 曲（表 1）を用いた．これらの楽曲には，市販 CD と同様に，ドラム音だけでなくさまざまな楽器音やボーカルが含まれている．すべてのデータは 16bit 量子化，44.1kHz サンプリングされている．

正解条件は，検出された発音時刻と実際の発音時刻とのずれが 25 [ms] 以下であることとした．実際の発音時刻 (ground truth) は，音響信号とあわせて配布されている標準 MIDI ファイルからバスドラム音とスネアドラム音の発音時刻を読み出し，それらを音響信号に手作業で同期させることで得た．得られた発音時刻数はバスドラムで 20818，スネアドラムで 15228 であった．認識結果の評価は，再現率，適合率，F 値で行うものとし，それぞれ次式で算出する．

$$\text{再現率} = \frac{\text{正解した発音時刻数}}{\text{実際の発音時刻数}}$$

$$\text{適合率} = \frac{\text{正解した発音時刻数}}{\text{提案手法により検出された発音時刻数}}$$

$$\text{F 値} = \frac{2 \cdot \text{再現率} \cdot \text{適合率}}{\text{再現率} + \text{適合率}}$$

### 4.2 ドラム音認識実験

まず，テンプレート適応 (TA) とテンプレートマッチング (TM) に基づくドラム音認識部の性能を評価した．すなわち，表 1 にある楽曲の音楽音響信号を入力としてドラム音の発音時刻列 $T_B, T_S$ を出力し，再現率，適合率，F 値で評価した．このとき，テンプレート適応の有無で性能比較を行った．

利用した手法を表 2 に，認識結果を表 3, 4 の上 2 段に示す．TM に TA を組み合わせた場合，バスドラム音，スネアドラム音認識時の F 値はそれぞれ 76.8%, 78.0%であった．TA なしの場合と比べて F 値がそれぞれ 6.77%, 10.0%向上したことから，TA が有効に機能したことが分かる．この結果は，補正処理による認識率向上を考察する上でのベースラインとなる．

表 2　実験に用いる手法一覧

| | |
|---|---|
| TM | Template Matching |
| TA | Template Adaptation |
| ECM | Error Correction by Matching |
| ECA | Error Correction by Adaptation |
| ECAM | Error Correction by Adaptation and Matching |

### 4.3 テンポ推定実験

ドラム音の発音時刻列をドラムパターンに分割する(3.3.2 節参照)とき，副次的な効果としてテンポ推定を行うことができる．4 拍長のリファレンスパターン $R_B, R_S$ を切り出しの指標としたので，切り出されたドラムパターンは 4 拍であるとして bpm を算出した．つまり，切り出されたドラムパターン中で最頻の時間長を $l_m$ [frames] とすると，bpm $= \frac{6000}{l_m} \cdot 4$ となる．

実験結果を表 5 に示す．50 曲中 40 曲に対して正しくテンポが推定できた．倍速誤りは切り出されたドラムパターンの時間長が実際には 2 拍であったことに起因する．正解と倍速誤りの楽曲は合計 98%を占め，ドラム音の発音時刻列がテンポ推定の有用な手がかりとなりえる．今後，ドラムパターン内の発音時刻数をモデル化するなどの倍テンポ誤り補正が課題である．

### 4.4 認識誤り補正実験

最後に，ドラム音の認識誤り補正手法（ECAM）の性能を評価した．認識誤り補正は，ドラムパターン推定に基づき信頼度付与された時刻をドラム音認識部の TA と TM にフィードバック情報として与えることで行う．比較のため，TA へのフィードバックのみによる補正（ECA）と TM へのフィードバックのみによる補正（ECM）の性能も評価した．

利用した手法を表 2 に，認識結果を表 3, 4 の下 3 段に示す．提案する ECAM によって，バスドラム音認識時の F 値は 76.8%から 81.1%（エラー削減率 18.7%）へ，スネアドラム音認識時の F 値は 78.0%から 80.3%（エラー削減率 10.6%）へ向上した．また，ECA，ECM を利用すると認識率が向上し，これらを組み合わせた ECAM ではさらに認識率が向上したことから，TA と TM へのフィードバックがともに有効であることが分かる．バスドラム音認識時は TA へのフィードバックがより有効であり，スネアドラム音認識時はほぼ同じであった．

表 3　50 曲を対象としたバスドラム音認識実験結果

| | 再現率 | 適合率 | F 値 |
|---|---|---|---|
| TM | 70.122% | 70.109% | 70.115% |
| TM+TA | 75.838% | 77.758% | 76.786% |
| TM+TA+ECM | 76.194% | 78.872% | 77.510% |
| TM+TA+ECA | 79.691% | 81.463% | 80.567% |
| TM+TA+ECAM | **79.835%** | **82.449%** | **81.121%** |

表 4　50 曲を対象としたスネアドラム音認識実験結果

| | 再現率 | 適合率 | F 値 |
|---|---|---|---|
| TM | 67.126% | 68.891% | 67.997% |
| TM+TA | 77.968% | 78.025% | 77.996% |
| TM+TA+ECM | 78.106% | 80.747% | 79.404% |
| TM+TA+ECA | 78.191% | 80.523% | 79.340% |
| TM+TA+ECAM | **78.283%** | **82.464%** | **80.319%** |

表 5　50 曲を対象としたテンポ推定実験結果

| 正解 | 倍テンポ誤り | それ以外 |
|---|---|---|
| 40 曲 (80%) | 9 曲 (18%) | 1 曲 (2%) |

## 5. おわりに

本稿では，音楽音響信号を入力としてドラム音の発音時刻列を出力し，ドラムパターン推定結果に基づいて認識誤り補正を行う手法について述べた．本手法は，音楽音響信号中のドラム音を認識し，ドラムパターンを推定するボトムアップ処理と，ドラムパターンを制約としてドラム音認識誤りを補正するトップダウン処理からなる．これらの処理は別々に存在せず，認識部の出力に対するドラムパターン推定結果を，認識部へフィードバックさせて再認識を行うことで補正部を実装したことが本手法の独創的な点である．

ドラムパターン中の認識誤りの可能性が高い時刻を検出するため，同じドラムパターンは連続して反復されやすいという仮定に基づく単純なドラムパターン遷移モデルを考案した．現実には，厳密にこのモデルに従う楽曲区間は多くないが，50 曲のポピュラー音楽を対象とした認識誤り補正実験では認識率向上を確認した．このことは，このモデルが現実のドラムパターン遷移をある程度反映していることを示唆している．

今後の課題として，ドラムパターン遷移モデルを洗練化することが挙げられる．ドラムパターンの分布や遷移は楽曲のリズムと密接な関係があると考えられる．そのため，MIR を実現する上で不可欠なリズム理解の手がかりに利用していく予定である．

謝辞 本研究は，科研費 (A) No.15200015，日本学術振興会特別研究員 (DC1) 科研費の補助を受けた．

## 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, 音響誌, Vol. 60, No. 11, pp. 675–681 (2004).
2) Yoshii, K., Goto, M. and Okuno, H.: Automatic Drum Sound Description for Real-World Music Using Template Adaptation and Matching Methods, *ISMIR*, 184–191 (2004).
3) Goto, M.: An Audio-based Real-time Beat Tracking System for Music With or Without Drum-sounds, *J. of New Music Research*, Vol. 30, No. 2, 159–171 (2001).
4) Gouyon, F. and Dixon, S.: A review of automatic rhythm description systems, *CMJ*, Vol. 29, No. 1 (2005).
5) Gouyon, F. and Herrera, P.: Determination of the meter of musical audio signals: Seeking recurrences in beat segment descriptors, *Proc. 114th AES Convention* (2003).
6) Scheirer, E.: Tempo and Beat Analysis of Acoustic Musical Signals, *JASA*, Vol. 103, No. 1, 588–601 (1998).
7) Otsu, N.: A Threshold Selection Method from Gray-Level Histograms, *IEEE Trans. SMC*, Vol. 6, No. 1, 62–66 (1979).
8) 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: 研究目的で利用可能な著作権処理済み楽曲・楽器音データベース, 情処学論, Vol. 45, No. 3, 728–738 (2004).