2X-8

音楽と自分の声を聞き分けながら ビートに合わせて発声するロボットの開発

水本 武志 † 武田 龍 ‡ 吉井 和佳 ‡ 駒谷 和範 ‡ 尾形 哲也 ‡ 奥乃 博 ‡

† 京都大学 工学部情報学科 ‡ 京都大学大学院 情報学研究科 知能情報学専攻

1. はじめに

我々の目的は, 自分の耳で音楽を聞き分け, それに合わせて発声するロボットを開発することである. 従来, ロボット自身の耳で音を聞くロボット聴覚の研究で, 複数の音声の混合音を聞き分けるロボットが多く開発されてきたが, ロボットが実世界の音をさらに処理するには, 音声以外の音の認識・理解が不可欠である.

我々は, ロボットの音楽理解能力は認識能力と表現能力から構成されると考える. つまり, ロボットによる音楽表現を外部から観察し, “ロボットは正しく認識して表現した” と判断できれば, ロボットが音楽を理解したと解釈する. この能力を実現する第 1 段階として, 本研究では, 認識能力として音楽を聞き分けながらビート構造を予測でき, 表現能力としてビートにあわせて「一, 二, 三, 四」と発声できるロボットを開発した.

2. 音楽に合わせた発声機能実現上の問題

これまでにも音楽を聞いて動作するロボットは複数開発されてきたが, それらはロボット自身が生成した音への対処はしていない. 琴坂らは神経振動子を用いて人間の演奏に同期した打楽器演奏をするロボットを開発した [1]. 対象はポピュラー音楽などの多重奏ではなく, ロボットの演奏をロボット自身が聞く事を考慮していない. 吉井らはポピュラー音楽からビート構造を予測し, 曲のテンポが変化しても音楽に合わせてステップを踏める機能を実現した [2]. 村田らはさらにハミング機能を追加したが, ハミングによるビート認識精度の低下は無視できないと報告している [3]. 従って, ロボットは自発声を抑圧し, 音楽を聞き分ける必要がある.

問題設定
目標: ロボットが自分自身の耳で音楽を聞き,
　　音楽のビートに合わせて発声するシステム
入力: 音楽音響信号と自発声の混合音
　　仮定: 自発声は波形が既知
出力: 音楽音響信号のビートに合わせた発声

本問題を自発声の抑圧, 音楽音響信号からのビート予測, 発声制御の 3 つの課題に分割する.

課題 1: 自発声の抑圧

自分が発声した音は, 前述したようにビート認識に大きく影響を及ぼすので, 自発声の抑圧は不可欠である. 本課題では, 自発声は既知なので, 未知の音楽と既知の発声の混合音から未知音楽を抽出する問題と設定できる.

課題 2: 音楽音響信号からのビート 予測

ビート予測のためには, 音楽音響信号からビート抽出を行い, それを基にビート構造を認識する必要がある. ここで, ビート構造とは, 4 分音符レベル, 2 分音符レベル, 小節レベルからなるビートの階層構造のことである.

なお, 音楽音響信号がどのような楽器を含むかは事前に分からないので, 楽器ごとの音源分離は仮定しない.

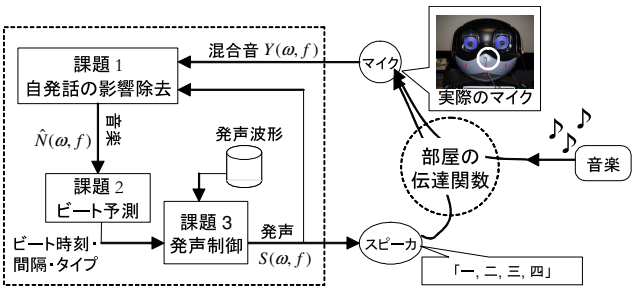

図 1: システム構成図と Robovie R2 のマイク

課題 3: 発声制御

認識し, 予測したビート構造は, 「一, 二, 三, 四」という発声により表現する. このような構造を含む表現により, [2] のようなビート間隔, すなわち音楽のテンポのみに基づく表現よりも豊富な情報が発現できる.

予測したビートと発声のタイミングの同期は重要な問題である. なぜなら, 発声内容によってそのアクセントの位置は異なるので, 全て同じタイミングで発声させるとビートに合う発声ができなくなってしまうからである. 特に, 複雑な発声内容 (例：歌詞) では顕著である.

3. 3 つの課題の解決策

開発したシステムの構成図を図 1 に示す.

3.1 解決策 1 : 適応フィルタによる自発声抑圧

雑音に適応的かつ頑健な既知信号の抑圧ができる独立成分分析に基づく適応フィルタ [4] を用いる. 混合音の分離過程を短時間フーリエ解析 (STFT) 後のスペクトルを用いて式 (1) のようにモデル化する.

$$\begin{pmatrix}\hat{N}(\omega,t)\\ \mathbf{S}(\omega,t)\end{pmatrix} = \begin{pmatrix}a(\omega) & -\mathbf{w}^T(\omega)\\ \mathbf{0} & \mathbf{I}\end{pmatrix}\begin{pmatrix}Y(\omega,t)\\ \mathbf{S}(\omega,t)\end{pmatrix} \quad (1)$$

$$\mathbf{w}(\omega,t) = [w_1(\omega), w_2(\omega), \cdots, w_M(\omega)]^T$$

$$\mathbf{S}(\omega,t) = [S(\omega,t), S(\omega,t-1), \cdots, S(\omega,t-M)]^T$$

$\mathbf{S}(\omega,t), Y(\omega,t), \hat{N}(\omega,t)$ はそれぞれ, 自発声, 混合音 (マイク入力) , 分離音 (推定された音楽) のスペクトルである. ω は周波数, t はフレーム, $\mathbf{w}$ は分離フィルタ, $a(\omega)$ はスケーリング係数, $\mathbf{I}$ は M 次単位行列である.

学習アルゴリズムを以下に示す.

$$\hat{N}(t) = Y(t) - \mathbf{Y}(t)^T\mathbf{w}(t)$$

$$\mathbf{w}(t+1) = \mathbf{w}(t) + \mu_1\phi(a(t)\hat{N}(t))\mathbf{S}(t)$$

$$a(t+1) = a(t) + \mu_2[1 - \phi(a(t)\hat{N}(t))a(t)\hat{N}(t)]a(t)$$

μ_1, μ_2 は学習係数, ϕ は非線形関数である. 音楽は優ガウス分布に従うので $\phi(y_i) = tanh(|y_i|)e^{j\theta(y_i)}$ を使う. 表記を簡単にするため, 周波数 ω は省略した.

Development of a robot capable of counting the beats by separating music and recognizing beats from a mixture of music and its own counting voice: *Takeshi Mizumoto, Ryu Takeda, Kazuyoshi Yoshii, Kazunori Komatani, Tetsuya Ogata, and Hiroshi G. Okuno (Kyoto Univ.)*

3.2 解決策 2 : リアルタイムビートトラッキングによるビート認識

ビートの認識にはビート構造が予測できるリアルタイムビートトラッキングシステム (RBTS)[5] を用いる. RBTS の入力は自発声が抑圧された音楽音響信号 $\hat{N}$ である. 入力音楽は 4/4 拍子であり, テンポ変動は大きくないと仮定している. 出力は 4 分音符単位の予測結果であり, ビート時刻とビート間隔, ビートタイプで構成される. ビートタイプはビート構造に置けるビートの位置, すなわち小節の何番目の 4 分音符かを表す.

ビート構造は次のように決定される. まず, 音響信号から 4 分音符単位でビートを認識する (4 分音符レベル). 次に, 強拍か弱拍かを判断する (2 分音符レベル). そして, 小節の先頭か否かを判断する (小節レベル).

ビート予測の基礎となるのが発音時刻である. 発音時刻とは音響信号の立ち上がり成分のピークである. 立ち上がり成分 $d(\omega,t)$ は分離音のパワースペクトル $p(\omega,t) = |\hat{N}(\omega,t)|$ を用いて次のように定義される.

$$pp = max(p(\omega,t-1), p(\omega,t-\pm 1)) \quad (2)$$

$$pp < min(p(\omega,t), p(\omega,t+1)) \quad (3)$$

$$d(\omega,t) = \begin{cases} max(p(\omega,t), p(\omega,t+1)) - pp & ((3)\ 成立) \\ 0 & (otherwise) \end{cases}$$

t はフレーム, ω は周波数である. また, 窓長は 4096 point (93 msec), シフト幅は 512 point (12 msec), サンプリング周波数は 44.1 [kHz] とする.

3.3 解決策 3 : 発声制御

認識結果は, 予測されたビート時刻に発声することで表現する. さらに, 小節の先頭なら「一」, 2 番目なら「二」, 以下「三」「四」と発声させることでビート構造も表現する. 発声は事前に録音した.

発声ごとのタイミング制御の問題は. 事前に発声の発音時刻の 3.2 の方法で求め, この時刻とビート時刻が合うように発声することで解決した. これにより, RBTS におけるビートの意味で音楽と合った発声ができる.

予測の遅れによって RBTS からの出力が得られない場合に発声が途切れるのを防ぐ必要がある. その場合は, 新しいビートの予測が得られるまでは前回の予測結果に基づいて発声を行うようにした. 例えば, 前回の発声が「一」なら, 予測結果が得られなくても前回と同じビート間隔だけ待った後に「二」を発声する.

4. ビート認識能力の評価実験

4.1 実験環境

ロボットには Robovie-R2 を使用した. 音楽は RWC 音楽データベース [6] のポピュラー音楽からテンポの異なる 3 曲 (RWC-MDB-P-2001 No.52, No.94, No.56) を 1 分ずつ使用した. 音楽音源とマイクの距離は約 1.4m, 自発声音源とマイクの距離は約 40cm とした.

ビート認識は, 発声と音楽の混合音, 混合音から発声を抑圧したもの, 発声なしの 3 条件で行った. また, 発声タイミングをランダムにし, 発声のテンポがずれた場合の影響も調べた. また, すべての処理はオンラインで行った.

4.2 実験結果と考察

RBTS の結果を図 2 に示す. 図より, 自発声の抑圧なしでは, 1 曲目開始時点で追従が遅れ, さらに, 3 曲目開始後に追従に失敗する. つまり, 自発声の抑圧しない場合はテンポ変化点で性能が悪化している. それに対して抑圧した場合は, テンポ変化に早く追従できるので, 自発声の抑圧は RBTS の性能の悪化を防げることがわかる.

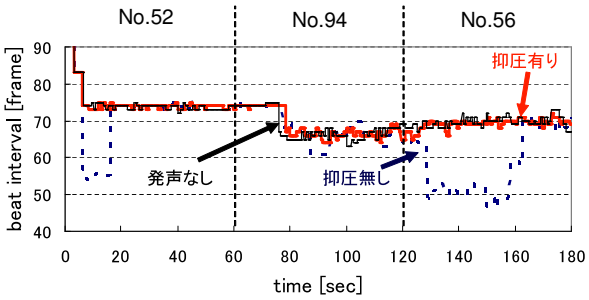

図 2: 自発声の抑圧によるビート認識能力の変化

2 曲目開始時に示すように, 自発声を抑圧した場合は発声しない場合に比べるとトラッキングの開始時刻は約 1.9 [sec] 遅い. 原因は, 本研究では考慮しなかった録音ノイズや背景ノイズ, 反響などが考えられる. しかし, 発声しない場合でも 2 曲目でテンポ変化への追従が 16 [sec] 遅れるので, RBTS 自体の性能の改善も必要である.

なお, データは示さないが, 発声のタイミングをランダムにした場合も, 自発声の抑圧によりビート認識能力が改善されることを確認した.

5. まとめ

本稿では, ロボットによる音楽理解を目指して, 音楽に合わせて発声するロボットを開発した. 音楽理解能力は認識能力と表現能力の 2 つから構成されると考え, 前者の能力は RBTS を, 後者の能力はビート構造に合わせた発声で実現した. その際, 自発声によってビート認識能力が低下する問題があったが, 適応フィルタによって自発声を抑圧することで解決した. その結果, テンポ変化におけるビート認識能力の改善を確認した.

認識能力の向上には, 実環境に起因する音楽の反響への対処や, テンポ変化への追従速度の改善, やビート予測精度の精度の改善が必要である. 表現能力の向上には, ビート構造の発声による表現を挙動に置き換えたダンスが考えられる. 動作で表現する場合は物理的な制約による遅延が大きいので補正が必要がある. また, 同じ発声による表現でも歌を使うときは歌とビートのアラインメントを考える必要がある. 両者の統合として, 表現した結果を認識能力へフィードバックして認識能力を改善するという方法も考えられる.

また, 本手法の汎用性を評価するために HRP-2 等のロボットによる挙動を含めた評価実験も行う予定である.

謝辞 本研究の一部は, 科研費, GCOE の支援をうけた.

参考文献

[1] S.Kotosaka *et al.* 神経振動子を用いたロボットのリズミックな動作生成, 日本ロボット学会誌, Vol. 19, No. 1 pp. 116-123, 2001

[2] K. Yoshii *et al.* A Biped Robot that Keeps Steps in Time with Musical Beats while Listening to Music with Its Own Ears. IROS,1743-1750, 2007

[3] 村田他 ボットによるビートトラッキングにおける周期性自己発声音の影響評価, SI, 2007, 1258-1259

[4] 武田他 ロボット音声対話のための MFT と ICA によるバージイン許容機能の評価, 情処第 70 回全大, 2008

[5] M. Goto An Audio-based Real-time Beat Tracking System for Music With or Without Drum-sounds, Journal of New Music Research, pp. 159-171, 2001

[6] 後藤他 RWC 研究用音楽データベース: ポピュラー音楽データベースと著作権切れ音楽データベース, 情処 音情研, 2001 巻, 35-42, 2001