人・ロボットインタラクションに向けた ビートトラッキングロボットの開発とその評価

村田 和真[1], 中臺 一博[1 3], 武田 龍[2], 吉井 和佳[2], 奥乃 博[2], 鳥井 豊隆[3], 長谷川 雄二[3], 辻野 広司[3]
[1] 東京工業大学大学院 情報理工学研究科, [2] 京都大学大学院 情報学研究科
[3] （株）ホンダリサーチ・インスティチュート・ジャパン

A Beat-Tracking Robot for Human-Robot Intraction and Its Evaluation

Kazumasa MURATA[1], Kazuhiro NAKADAI[1 3], Ryu TAKEDA[2], Kazuyoshi YOSHII[2],
Hiroshi G. OKUNO[2], Toyotaka TORII[3], Yuji HASEGAWA[3], Hiroshi TSUJINO[3]
[1] Tokyo Institute of Technology, [2] Kyoto University,
[3] Honda Research Institute Japan Co., Ltd.

Abstract— Human-robot interaction through music in real environments is essential for humanoids, because such a robot makes people enjoyable. We thus developed a beat-tracking robot which steps, sings, and scats according to musical beats predicted by using its own ears. This paper first describes the beat-tracking robot, and then evaluated it in detail in the following three points: adaptation to tempo changes, robustness of environmental noises including periodic noises generated by stepping, singing and scatting, and human-robot interaction by using a clapping sound. The results showed that our beat-tracking robot improved drastically in comparison with the reported one with a well-known real-time beat-tracking algorithm.

Key Words: beat tracking, human-robot interaction, noise robustness, robot audition

1. はじめに

近年，人とソーシャルなインタラクションを行うロボットの研究が多く行われている．人が演奏した音楽に応じて歌ったり踊ったりする機能は，人とより自然で豊かなインタラクションを行うために重要な機能である．我々は音楽を自分の耳で聞き，リズムを抽出して足踏みをするロボットを開発してきた [1]．本稿では実環境でのロバスト性を向上させるために，ビート変化に対する追従遅れを低減する STPM(Spectro-Temporal Pattern Matching) と，自己発声音を抑制するためのセミブラインド独立成分分析 (セミブラインド ICA) の二つを導入し，足踏みと同時にスキャット[1]や歌唱のできるビートトラッキングロボットを開発した．本稿では，まずこのビートトラッキングロボットについて説明し，次にテンポ変化に対する追従性，周期ノイズを含んだ環境ノイズに対するノイズロバスト性及び手拍子を用いた人とロボットのインタラクションを評価した．

2. STPM ベースのビートトラッキング

STPM は時間周波数領域でパターンマッチングを行うビートトラッキング手法であり，一般的な自己相関関数を用いるよりも短い窓長でロバストにテンポを抽出する事が出来る．このビートトラッキングアルゴリズムは「周波数解析」「テンポ予測」「ビート時刻予測」の３つのモジュールからなっており，入力信号に対してビート時刻とテンポを出力する．

[1]本稿はビートにあわせて「ずん」「ちゃ」と発声する事をスキャットと呼ぶものとする．

2·1 周波数解析

44.1kHz で同期してサンプリングされた２種類の入力信号に対して周波数解析を行う．入力信号のうち一つはロボットのマイクで録音した音楽音響信号で，一般に雑音が混入している．もう一つはスキャットや歌声などのロボットの自己発声音で，既知信号であるためクリーンな信号が利用可能である．この信号は後述する雑音抑制で用いる．周波数解析では２つの信号それぞれに対して短時間フーリエ変換 (STFT) を行う．ここで，窓関数には窓長 4096 ポイントのハニング窓を用い，シフト長は 512 ポイントとした．次にスキャットや歌声などの自己発声音のエコーキャンセルを行う．歌声やスキャット，足踏みによる駆動音などはビートと同様に周期性を持つノイズとなるため，ビート予測精度の低下やビート変化に対する追従遅れを生じさせる原因となるからである．具体的な雑音除去にはセミブラインド ICA[2] に基づくエコーキャンセル法を使用する．この手法は一般的な適応フィルタベースの手法より性能が高いことが報告されておりマルチチャンネル処理にも簡単に拡張できるという特徴を備えている．本稿では前述の２つの入力信号のスペクトルに対し，この手法を適用した．まず，音声認識や音楽認識で用いられるメルフィルタバンクを用いて 2,049 次元のスペクトルを 64 次元に圧縮した．メルスケールのスペクトログラム上でパワーが急激に上昇している時刻はオンセットである可能性が高いと考え，急激なパワー上昇をしている領域を見つけるためパワースペクトログラムに対してソーベルフィルターを適用してエッジの強調を行う．その後上昇部分のみを抜き出すことによってそ

れぞれの時間フレームで 62 次元のオンセット時刻ベクトル $d(t,f)$ が得られる．ここで t は時間フレーム，f はメルフィルタバンクの次元を表す．

2·2 ビート間隔推定

隣り合う二つのビートの間隔をビート間隔と定義し，その長さを推定する．まず，オンセット時刻ベクトルを用いて時間-周波数領域でパターンマッチングを行う．パターンマッチング関数として以下で定義される Normalized Cross-Correlation (NCC) を用いる．ビート間隔信頼度 $R(t,i)$ は

$$R(t,i)=\frac{\sum_{j=1}^{62}\sum_{k=0}^{P_{width}-1} d(t-k,j)d(t-i-k,j)}{\sqrt{\sum_{j=1}^{62}\sum_{k=0}^{P_{width}-1} d(t-k,j)^2\cdot\sum_{j=1}^{62}\sum_{k=0}^{P_{width}-1} d(t-i-k,j)^2}} \tag{1}$$

ここで P_{width} はパターンマッチングの窓長で i はシフトパラメータである．.

提案するアルゴリズムは時間と周波数の情報を同時に利用している．このため窓長が 1 秒程度と短い場合でもロバスト性を保ったままテンポ変化に対して素早い追従が可能である．これは，従来用いていた後藤らのアルゴリズム [3] で，自己相関関数の窓長が約 6〜10 秒であることからも短いといえよう．

偽のビート間隔の検出を避けるために，まずビート間隔を我々のテストベッドロボットの動作範囲である 61M.M. から 120M.M.[2] に制限する．次に候補のインターバルを $R(t,i)$ のローカルピークを大きい方から I_1, I_2 の二つ選ぶことによって得る．裏拍などの影響で 2 倍 3 倍のビート間隔が間違いとして検出される傾向を考慮し，本来のビート間隔は以下のように推測する．

$$I(t)=\begin{cases} 2|I_1-I_2| & (|I_{n2}-I_1|<\delta \text{ or } |\mathrm{I_{n2}-I_2}|<\delta) \\ 3|I_1-I_2| & (|I_{n3}-I_1|<\delta \text{ or } |\mathrm{I_{n3}-I_2}|<\delta) \\ I_1 & otherwise, \end{cases} \tag{2}$$

$$I_{n2}=2|I_1-I_2|,\quad I_{n3}=3|I_1-I_2|,$$

ここで δ はエラーパラメータである．

2·3 ビート時刻予測

まずビート時刻信頼度を近接ビート信頼度 $S_c(t,i)$ と連続ビート信頼度 $S_r(t,i)$ を用いて以下のように定義する．

$$S(t)=\sum_i S_c(t-i,i)S_r(t-i,i) \tag{3}$$

近接ビート信頼度とはある時刻とビート間隔前の時刻がともにビート時刻である尤度のことである．現時刻とそのビート間隔前の時刻のオンセット時刻ベクトルを用いて計算する．時刻 t において時刻 $t-i$ に存在するビートの近接ビート信頼度 $S_c(t,i)$ は以下のように表せる．

$$S_c(t,i)=\begin{cases} F_s(t-i)+F_s(t-i-I(t)) & (i\le I(t)) \\ 0 & (i>I(t)) \end{cases} \tag{4}$$

$$F_s(t)=\sum_{f=1}^{62} d_{inc}(t,f)$$

連続ビート信頼度とはビート間隔でビートが続いているかの信頼度である．近接ビート信頼度を用いて以下のように定義した

$$S_r(t,i)=\sum_m^{N_{S_r}} S_c(T_p(t,m),i) \tag{5}$$

$$T_p=\begin{cases} t-I(t) & (m=0) \\ T_p(t,m-1)-I(T_p(t,m)) & (m\ge 1) \end{cases}$$

ここで $S_r(t,i)$ は時刻 t における時刻 $t-i$ に存在するビートに対する連続ビート信頼度を意味する．$T_p(t,m)$ は時刻 t を基準として m 個前のビート時刻で，N_{S_r} は連続ビート信頼度を評価する際のビート数である．この信頼度は複数のビート列が見つかった場合に，一番強いビート列を求める際に用いる．

次にビート時刻を推定する．まず，n 番目のビート時刻を $T(n)$ とする．$T(n)\ge t-\frac{3}{4}I(t)$ の時，区間 $[T(n)+\frac{1}{2}I(t), T(n)+\frac{3}{2}I(t)]$ 内で上位 3 つのピークを抽出する．次に $T(n)+I(t)$ に一番近いピークを次のビート時刻 $T(n+1)$ とする．ピークが区間 $[T(n)+\frac{2}{3}I(t), T(n)+\frac{4}{3}I(t)]$ に存在しない場合，$T(n)+I(t)$ を $T(n+1)$ とする．

ビート時刻 $T(n+1)$ は現在より前の時刻のビート，つまり $t>T(n+1)$ である．スキャットや歌機能を実現するためには未来のビート時刻 T' を予測する必要がある．以下の外挿を用いて，未来のビート時刻を予測する．

$$T'=\begin{cases} T_{\mathrm{tmp}} & \text{if } \mathrm{T_{tmp}}\ge\frac{3}{2}\mathrm{I_m(t)+t} \\ T_{\mathrm{tmp}}+I_m(t) & \text{otherwise.} \end{cases} \tag{6}$$

$$T_{\mathrm{tmp}}=T(m)+I_m(t)+(t-T(m))-\{(t-T(m)) \mod I_m(t)\}$$

ここで $I_m(t)$ は $I(t)$ 群の中央値で，$T(m)$ は検出したビート時刻の中で一番新しい時刻である．

3. 実装

Fig. 1 に我々のビートトラッキングロボットのアーキテクチャを示す．このシステムは主に三つのサブシステム分けることができる．人型ロボットとリアルタイムビートトラッカーとロボット制御である．

Honda ASIMO をテストベッドとして用いた．また本稿では，ロボット頭部に搭載されているマイクのうち 1 本のみを利用した．このロボットのステップ可能な間隔は 1,000 から 2,000 [ms] の範囲となっている．これは音楽のテンポに換算すると 61〜 120 M.M. である．なお，発声用のスピーカは ASIMO の胸部に内蔵されている．

リアルタイムビートトラッカーはロボットのマイクロフォンから入力された音楽音響信号を用いてビート間隔とビート時刻を計算し出力する．このときスキャットや歌声といった出力信号を同時に入力することで音楽音響信号に混入した自己発声音をキャンセルする．

[2] Mälzel's Metronome: 一分あたりの四分音符の数．たとえばテンポ 60 M.M. であれば，四分音符の長さは 1,000 [ms] である．

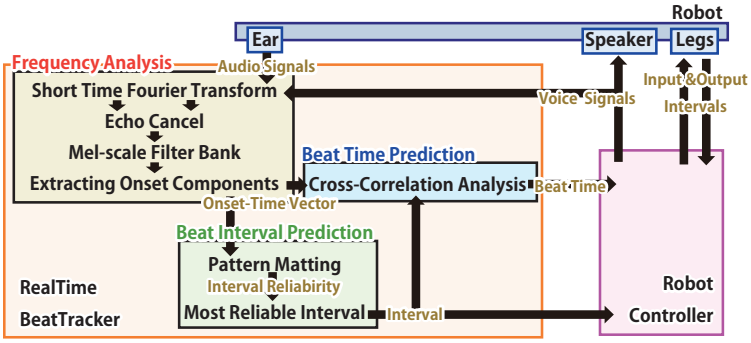

Fig.1 Architecture of a beat tracking robot

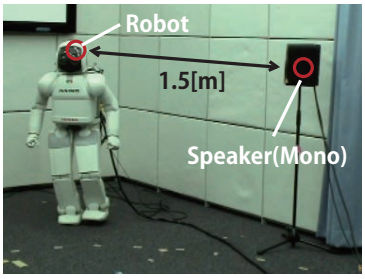

Fig.2 Experimental condition

ロボット制御はリアルタイムビートトラッカーによって検出されたビート時刻とビート間隔にあわせて，ステップ、スキャット、歌唱を行うため，ある程度の分散をもった遅延が発生する．

ロボットの足踏みに関しては [1] と同様に足の接地時刻とビート時刻、ビート間隔を用いた簡単なフィードバック則を用いて制御した．

スキャットは，ビート時刻に合わせて「ずん」「ちゃ」が自然に聞こえるよう「ずん」「ちゃ」の各音とビート時刻の同期のタイミングが重要である．そこで「ずん」，「ちゃ」から抽出したオンセット時刻ベクトルの各値の合計値のピークを「ずん」「ちゃ」のビート時刻とする．この各音でのビート時刻と，音楽のビート時刻を合わせて発音する．

歌唱機能は，聴いた曲の速度とビートに同期して歌を歌うという機能である．今回はビート時刻ごとに音符の発音時刻と発音長 (duration)，発音記号をあらかじめ入力した楽譜（音階）と歌詞から計算し，MIDI 信号を出力する．この MIDI 信号を VOCALOID 2 に送出することによって歌機能を実装する．VOCALOID 2 は MIDI を用いて自然な歌声を合成できる商用ソフトである．VOCALOID 2 は MIDI データを受け取ってから発音するまでに 200[*msec*] の遅れがある．それを考慮し，発音時刻の 200[*msec*] 前に MIDI データを送信するよう実装を行った．

4. 実験

ビートトラッキングロボットを以下の 3 点で評価した．

実験 1 テンポ変化への追従速度
実験 2 ビート予測のノイズロバスト性能
実験 3 人間の手拍子への追従性能

これらを評価するために以下の二種類の音楽信号を用いて実験を行った．

T1 テンポ変化を含む音楽音響信号
RWC 音楽データベース (RWC-MDB-P-2001)[4] から 3 曲 (No. 11, No. 18, No. 62) を選んだ．これらは市販のＣＤのように様々な楽器音と歌声を含んでおり，テンポはそれぞれ 90，112，81 M.M である．これらを No.18 - No.11 - No.18 - No.62 の順に 60 秒区切りでつなげることで 4 分の音楽信号を作成した．

T2 テンポの固定された音楽音響信号
No. 62 の MIDI データを用いて生成したテンポ一定の曲を用いる．ただし，MIDI のデータはビート時刻の検証のみに用いる．

実験はすべて残響時間 0.2 秒 (RT_{20})， 4 m×7 m の部屋で行い，音楽用音源にはスピーカー (GENELEC 1029A) を用いた．ロボットとスピーカの距離は 1.5 m である (Fig. 2)．

実験１では **T1** を用いて 5 種類の状態でビートトラッキングの遅れを計測した．ビートトラッキングの遅れは実際にテンポが変化してからシステムがテンポ変化に追従するまでの時間の差とする．５種類のうち２種類は ASIMO の電源を off にしてスキャットを行った場合と行わなかった場合，３他の種類は ASIMO の電源を on にしてステップを踏みながらスキャットを行った場合，行わなかった場合，歌った場合である．

実験２では **T2** を用いて 5 種類の状態でビート推定の成功率を計測した．ビート推定の成功率 r は以下のように定義する．

$$r = \frac{N_{\mathrm{success}}}{N_{\mathrm{total}}} \times 100. \tag{7}$$

ここで N_{success} は推定が成功したビート数で，N_{total} は正解ビート総数である．推定されたビート時刻と正解ビート時刻の差が $\pm 100ms$ 以内に収まっている場合は推定が成功したものとした．これは過去の時間知覚の研究において立ち上がりタイミングが $\pm 100ms$ 以上の場合は完全に二音がずれて感じられることが知られているからである [5]．５種類のうち３種類は ASIMO の電源を切った状態で，１種類はスキャットなしでエコーキャンセルあり，残りの２種類は共にスキャットありだが，エコーキャンセルの有無に違いがある．他の２種類は ASIMO の電源を入れステップを踏みながらスキャットを行っている状態でエコーキャンセルを行った場合と行わなかった場合である．

実験３では最初被験者にロボットが手拍子に沿ってスキャットを行うことと 60bpm〜120bpm の間しか検出できないことのみを伝え，ロボットの前で 1 分間手拍子を行ってもらう．実験終了後「ビートが検出されていると思うか」「ビート変化に追従していると思うか」という設問に 5 段階で評価してもらった．次にロボットがスキャットに加えてステップも踏むことを伝え同じ実験を行った．実験終了後に同じ質問を行った．評価はスキャットの手拍子に対する再現率と適合率で行い，一致の判定は Ex.2 と同様 $\pm 100ms$ 以内の場合に行った．手拍子の総数を A，手拍子とスキャットが一致した数を

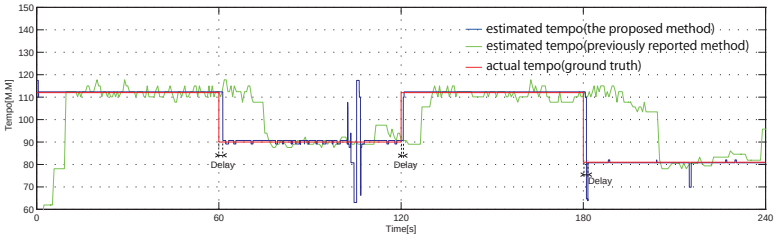

Fig.3 Result of using music of tempo chage.

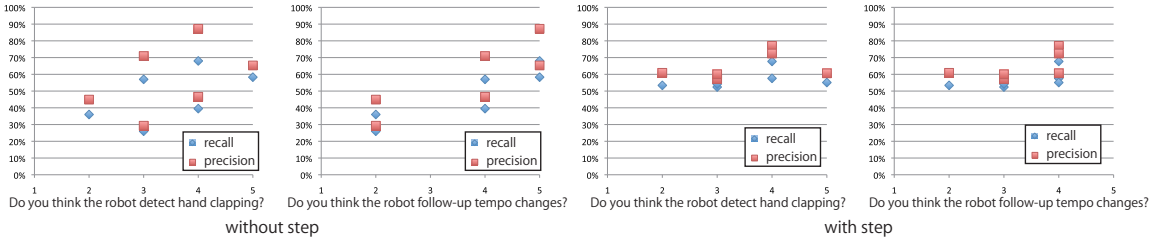

Fig.4 Result of using hand clapping.

C，スキャットの総数を N としたとき，再現率は $\frac{C}{A}$，適合率は $\frac{C}{N}$ である．

4·1　結果と考察

Fig. 3 は**実験１**の結果を表している．提案手法は従来手法に比べテンポ変化への適応が速いことがわかる．100 秒近くでビートトラッキングが乱れているのは **T1** がビート時刻にオンセットがない部分が一時的に存在しているからである．このため提案手法では一時的に，従来手法では長期に乱れる．遅れの平均値を Table 1 に示す．我々の提案した手法が従来手法よりスキャットのない場合で１０倍程度，ある場合で２０倍程度追従が高速であることがわかる．

Table 2 は**実験２**の結果を表している．“correct” はビートトラッキングシステムが正しいビートを予測した事を示し，“half shifted” は裏拍を予測した事を示す．これは自己発声音がその周期性のためにビートトラッキングに影響を与えていること，およびエコーキャンセルが影響を劇的に減らしていることを示している．また実際にスキャットや歌いながら音楽のビートに合わせてロボットがステップを踏むことが出来ることも確かめられた．

Fig. 4 からステップなしではアンケートの評価と再現率，適合率が比例している事がわかる．しかし，ステップがある場合には比例関係が認められない．これは足踏みが追加されたことによって，足踏みに引き込まれビートの一致の評価がわかりにくくなったことが考えられる．実際にアンケートにおいて引き込まれる感じがしたという回答を得ている．

5.　結論

ビートトラッキングロボットを実現するために，実時間性・安定性・テンポ変化への追従性を備えた時間周波数パターンマッチングベースのビートトラッキング手法を提案した．また，スキャットや歌などの周期性を伴った自己発声音の影響を削減するためにセミブラインド ICA によるエコーキャンセルを行った．さらに提案した手法を Honda ASIMO に実装し，ロボット搭載マイクを用いて収録した信号から実時間でビートを抽出し，そのビート情報に基づき，足踏み，口ずさみ，歌唱を行うビートトラッキングロボットを構築した．評価実験を通じて，構築したビートトラッキングロボットは，高いノイズロバスト性と，テンポ変化への高い追従性を示すことを確認した．さらに，手拍子を通じた人ロボットインタラクション実験からは，引き込み現象が発生していることが示唆された．ロボットの挙動の高度化，ロボット外の雑音源の抑制，画像など他のモダリティを用いたロバスト性の向上が今後の課題である．

Table 1 Tracking Delay for Tempo Changes

	ASIMO power off		ASIMO power on(with step)		
	w/o scatting	w/ scatting	w/o scatting	w/ scatting	w/ singing
reported	11.24	29.91	14.66	20.43	N/A
proposed	1.31	1.31	1.29	1.29	1.29

Table 2 Beat Prediction Success Rate

	ASIMO power off			ASIMO power on (with step)	
	w/o scatting	w/ scatting		w/ scatting	
		w/ echo cancel	w/o echo cancel	w/ echo cancel	w/o echo cancel
Correct	73%	76%	54%	77%	54%
Half shifted	3%	1%	22%	2%	28%

参考文献

[1] K. Yoshii *et al.*. A biped robot that keeps steps in time with musical beats while listening to music with its own ears. *IROS-2007*, pages 1743–1750, 2007.

[2] R. Takeda *et al.*. Exploiting known sound sources to improve ica-based robot audition in speech separation and recognition. *IROS-2007*, pages 1757–1762, 2007.

[3] M. Goto and Y. Muraoka. A real-time beat tracking system for audio signals. In *Proceedings of the International Computer Music Conference*, pages 171–174, San Francisco CA, 1995. International Computer Music Association.

[4] M. Goto *et al.*. RWC music database: Popular, classical, and jazz music databases. In *Int. Conf. Music Information Retrieval*, pages 287–288, 2002.

[5] R.A. Rasch. Synchronization in performed ensemble music. *Acustica*, 43(2):121–131, 1979.