5R-7

累積頻度重みを適用したパーティクルフィルタによる実時間楽譜追従

大塚 琢馬 † 中臺 一博 ‡ 高橋 徹 † 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 ‡(株) ホンダ・リサーチ・インスティチュート・ジャパン

1. はじめに

人間の演奏に呼応する計算機による自動伴奏システムや, 共演者音楽演奏ロボットの実現には, 人の演奏の音楽音響信号と, その楽譜の逐次的対応付け–楽譜追従– が重要な技術である [1, 2]. 従来の楽譜追従手法がよく使うのは楽譜の各和音を隠れ状態とした隠れマルコフモデル (HMM) である (例: Antescofo [3]). HMM に基づく楽譜追従手法は, 音響信号の調波構造などの音高情報の観測確率モデルと, 隣接和音の遷移確率モデルに基づいて音響信号に対する楽譜位置を逐次的に推定する. HMM の問題点は, 各状態の継続長が明示的にはモデル化されない点である. この結果, 音符長が安定せず楽譜位置推定が不正確になる. Antescofo は, 隠れセミマルコフモデル (HSMM) を用い, 各状態に留まることのできる制限時間を導入し, 一つの和音に留まり続けるという問題に対処している. しかし, HSMM には状態が速く進むことに対するペナルティがないので, 推定した楽譜位置が実際の演奏位置よりも前に進み過ぎるという問題が残っている.

この問題に対処するため我々は, 楽譜位置に加えてテンポ (曲の速さ) を同時に推定対象としパーティクルフィルタによる定式化による楽譜追従法を開発している [2]. テンポ推定により音符長が安定した結果, 楽譜追従精度が向上したものの, 楽譜位置推定に誤差が残っていた. この原因は, 音響信号観測モデルにおける音響信号・楽譜間調波構造類似度から観測重みへの変換時に, 類似度の差が観測重みにほとんど反映されていなかたためである. 本稿では, 過去の観測から得た調波構造類似度類似度の累積頻度ヒストグラムを用いた観測重みを導入した楽譜追従精度向上について報告する.

2. 手法の概要

まず, 本稿で扱う楽譜追従問題を定義する.

入力: 逐次的音楽音響信号と対応する楽譜,
出力: 音楽音響信号に対する楽譜位置とテンポ,
仮定: 楽譜は音階と音長を持つ (楽器は未知).

2.1 パーティクルフィルタによる楽譜追従

$X_{f,t}$ を入力音響信号を短時間フーリエ変換したとき, 周波数 f (Hz), 時刻 t (sec.) における複素振幅スペクトルとする. ただし, 我々の実装では音響信号のサンプリング周波数を 44100 (Hz), フーリエ変換の窓関数の長さを 2048 (pt), 窓関数のステップ幅を 441 (pt) で行うため, 周波数 f と時刻 t はそれぞれ, 21.5 (Hz), 10 (msec.) ごとに離散化される. k (beat) を四分音符の長さを単位とする楽譜位置インデックスとする. 楽譜は次のようにフレームに分割して扱う: 1 フレームの長さは四分音符の 1/12 の長さとし, 各フレームは存在する音の高さのリストを持つ.

パーティクルフィルタを用いた楽譜追従では, 音響信号スペクトログラム $\mathbf{X}_{0:n\Delta T} = [X_{f,0:n\Delta T}]$ が与えられたときの, 楽譜位置 k_n, ビート間隔 (テンポの逆数) b_n の確率密度 $p(k_n, b_n|\mathbf{X}_{0:n\Delta T})$ を推定する. ただし, n はフィルタリング処理のインデックス, ΔT (sec) はフィルタリングを行う時間間隔である. $\mathbf{X}_{0:n\Delta T}$ はフィルタリング開始時から現在時刻までの全周波数帯域のスペクトログラムを示す. パーティクルフィルタはこの分布を多数のパーティクルを用いて, 次の式 (1) ように近似して推定する.

$$p(k_n, b_n|X_{f,1:n\Delta T}) = \sum_{i=1}^{I} w_n^i \begin{bmatrix} k_n^i \\ b_n^i \end{bmatrix}, \quad (1)$$

ただし, i はパーティクルの添字, I は総パーティクル数, w_n^i, k_n^i, b_n^i はそれぞれ, n ステップ目でパーティクル i が持つパーティクル重み, 楽譜位置, ビート間隔である.

図 1 に 1 つのフィルタリングで行う処理を示す: (1) 提案分布からパーティクルのサンプリング, (2) 状態遷移モデル・観測モデルを用いたパーティクル重み計算, (3) 楽譜位置とビート間隔の点推定・リサンプリング.

(1) 提案分布からのサンプリング: 最新の観測スペクトログラムから, 音響信号と楽譜のオンセットが合うように n ステップ目の楽譜位置・ビート間隔 $[k_n^i\ b_n^i]$ をサンプルする (詳細は [2]).

(2) パーティクルの重み計算: 次式に従ってサンプルされた各パーティクルの重みを計算する.

$$w_n^i = \frac{p(k_n^i, b_n^i|k_{n-1}^i, b_{n-1}^i)\, p(\mathbf{X}_{\mathbf{T}_n}|k_n^i, b_n^i)}{q(k_n^i, b_n^i|\mathbf{X}_{\mathbf{T}_n})}. \quad (2)$$

$\mathbf{X}_{\mathbf{T}_n}$ は, 長さ L の時間窓 $\mathbf{T}_n = \{t|n\Delta T - L < t \leq n\Delta T\}$ で得た最新の観測スペクトログラムである. 式 (2) 分子の第 1 項は状態遷移モデル, 第 2 項は観測モデルと呼ぶ. また, 分母は k_n^i, b_n^i のサンプルされた提案分布である. 式 (2) による重み計算は重点サンプリングと呼ばれ, パーティクルの持つ値 k_n^i, b_n^i が状態遷移・観測モデルに合致するほど重み w_n^i は大きくなり, 提案分布からサンプルされやすい値ほど重みは小さくなる.

以下では紙面の都合上, 観測モデルに焦点を絞り, 詳細は文献 [2] にゆずる. 観測スペクトログラムの各フレームを確率密度とみなし, 楽譜から生成した調波 GMM との KL-Divergence (KL-Div) を算出する.

$$\tilde{k}^i(\tau) = k_n^i - \frac{n\Delta T - \tau}{b_n^i}, \quad (3)$$

$$D_{i,\tau}^{KL} = \int X_{f,\tau} \log \frac{X_{f,\tau}}{\hat{X}_{f,\tilde{k}^i(\tau)}} \mathrm{d}f. \quad (4)$$

ただし, 式 (3) で $\tilde{k}^i(\tau)$ は, 時刻 $n\Delta T$ の楽譜位置が k_n^i とし, ビート間隔 b_n^i のときの時刻 τ における楽譜位置である. 式 (4) における $\hat{X}_{f,\tilde{k}^i(\tau)}$ は楽譜から生成された調波 GMM である. 観測尤度は, 計算した各フレームの KL-Div を tanh 関数を用いて [0 1] 区間に正規化することで得る.

Particle Filter-based Real-time Score Following using Cumulative Similarity Histogram: *Takuma Otsuka (Kyoto Univ.), Kazuhiro Nakadai (HRI-JP), Toru Takahashi, Tetsuya Ogata, and Hiroshi G. Okuno (Kyoto Univ.)*

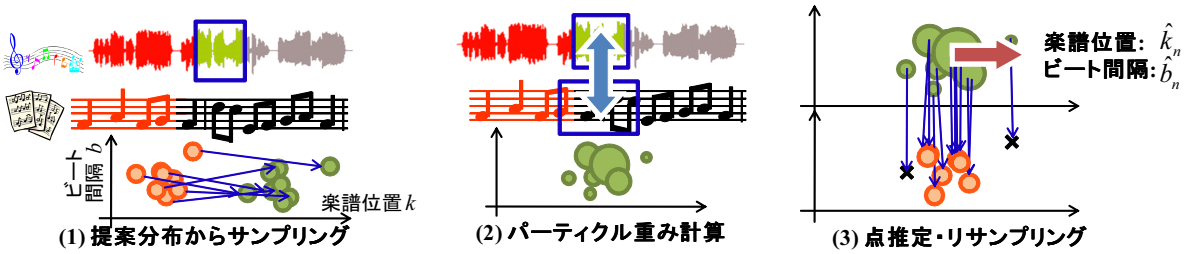

図 1: パーティクルフィルタ: 1 ステップで行う 3 つの処理

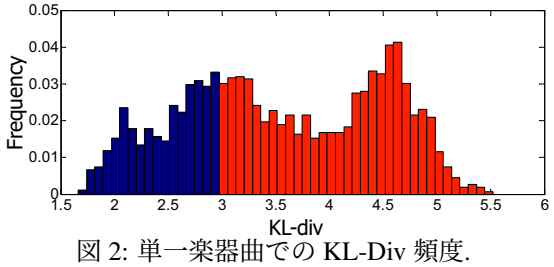

図 2: 単一楽器曲での KL-Div 頻度.

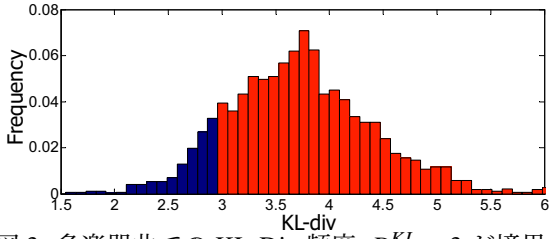

図 3: 多楽器曲での KL-Div 頻度. $D^{KL}_{i,\tau} = 3$ が境界.

$$p(\mathbf{X}_{\mathbf{T}_n}|k^i_n, b^i_n) = \frac{1}{L}\int_{\mathbf{T}_n}\left(\frac{1}{2}+\frac{1}{2}\tanh\frac{D^{KL}_{i,\tau}-\bar{D}^{KL}}{\nu}\right)\mathrm{d}\tau, \quad (5)$$

ただし, 実験的に $\bar{D}^{KL} = 4.2$, $\nu = 0.8$ とした.

(3) 楽譜位置とビート間隔の点推定・リサンプリング: 最終的な点推定値 $\hat{k}_n, \hat{b}_n$ は重みの大きなパーティクルが持つ値の重みつき平均として得る.

$$\left[\ \hat{k}_n,\quad \hat{b}_n\ \right] = \frac{\sum_{i\in I^{\mathrm{top}}_n} w^i_n \left[\ k^i_n,\quad b^i_n\ \right]}{\sum_{i\in I^{\mathrm{top}}_n} w^i_n}. \quad (6)$$

ただし, I^{top}_n は n 回目のフィルタリングで重みが上位 10% のパーティクル添字 i の集合である. リサンプリングは, 各パーティクルの重みに比例した確率で, I 個のパーティクルを独立にサンプルすることで行う.

2.2 KL-Div の累積頻度を用いた重み計算

KL-Div による調波構造類似度は楽器間の音量バランスや音色によって大きく変化する (図 2, 3). このため, 式 (5) による静的な尤度計算は, 場合によっては信頼性が低い. 本稿で示す手法を Algorithm 1 にまとめる. フィルタリングごとに式 (4) で計算する KL-Div の累積頻度を用いて観測尤度計算を行う. ただし, ある程度の KL-Div 値を得なければ高精度な尤度計算が行えないため, 最初の $N_{\mathrm{thres}} = 3$ 回のフィルタリングは従来の式 (5) による尤度計算を行う.

$$p(\mathbf{X}_{\mathbf{T}_n}|k^i_n, b^i_n) = \frac{1}{L}\int_{\mathbf{T}_n}\frac{\#(D^{KL}_{i,\tau} < \mathbf{D}^{KL})}{\#(\mathbf{D}^{KL})}\mathrm{d}\tau. \quad (7)$$

ただし, $\mathbf{D}^{KL}$ はこれまでの KL-Div の集合, $\#(D^{KL}_{i,\tau} < \mathbf{D}^{KL})$ は $\mathbf{D}^{KL}$ の中で $D^{KL}_{i,\tau}$ より大きな要素の個数であり, $\#(\mathbf{D}^{KL})$ は $\mathbf{D}^{KL}$ に含まれる要素数である. 図 2, 3 は式 (7) において, $D^{KL}_{i,\tau} = 3$ の時, 分子の $\#(D^{KL}_{i,\tau} < \mathbf{D}^{KL})$ を右側赤い領域, 分母の $\#(\mathbf{D}^{KL})$ を領域全体で表している.

Algorithm 1 累積頻度による観測尤度計算

```
D^KL ← {∅}
for i = 1 to I do
    式 (4) による D^KL_{i,τ} 計算 (τ ∈ T_n)
    if n > N_thres then
        式 (7): 累積頻度による尤度計算
    else
        式 (5): 従来法による尤度計算
    end if
    D^KL ← D^KL + {D^KL_{i,τ}} (τ ∈ T_n)
end for
```

表 1: 実験結果: 誤差 500 (msec.) 以下の区間の割合 (%)

	冒頭 30 秒	同 60 秒	全区間
従来法	63.5	52.6	33.2
本手法	69.3	56.5	36.2

3. 評価実験

本手法の有効性を, 式 (5) のみを尤度計算に用いる従来法と比較することで示す. 実験には, RWC 研究用音楽データベース [4] からジャズ楽曲 20 曲を用いた. 楽譜位置の逐次推定では, 楽曲が進むに連れて誤差が蓄積するという性質がある. その様子を示すため, 各楽曲の冒頭 30 秒, 60 秒, 全区間について, 楽譜位置推定誤差が 500 (msec.) 以下の区間を算出する. 各楽曲について得られた区間の割合の 20 曲分の平均値を表 1 にまとめる.

表 1 が示す通り, 誤差が小さく抑えられている区間は本手法の方が高く, 尤度計算法改善の効果が確認できる. ただし, 楽曲の経過につれて推定に誤差が蓄積され, 本手法の効果も薄れていく. そのため, 誤差が累積した状態からの復旧手法が今後の課題として残る.

4. おわりに

本稿では, 楽譜追従において楽譜からの音響信号尤度計算について, 計算された類似度の大小比を用いたパラメータチューニング不要な尤度計算法を示した. 従来の尤度計算法との比較実験では, 特に楽譜追従の初期段階の推定精度が改善されることが示された. 今後の課題としては, 蓄積する楽譜位置推定誤差への対処が挙げられる.

謝辞: 本研究は科研費 (S), GCOE の支援を受けた.

参考文献

[1] R. Dannenberg and C. Raphael: "Music Score Alignment and Computer Accompaniment", *Comm. ACM*, Vol. 49,No. 8, pp.38–43, 2006.

[2] T. Otsuka *et al.*: "Real-Time Audio-to-Score Alignment using Particle Filter for Co-player Music Robots", *EURASIP Journal of ASP*, vol. 2011, Article ID 384651, 2011.

[3] A. Cont: "A Coupled Duration-Focused Architecture for Realtime Music to Score Alignment", *IEEE Trans. on PAMI*, Vol. 32, No. 6, pp.974–987, 2010.

[4] M. Goto: "Development of the RWC Music Database", *Proc. of International Congress on Acoustics*, pp.I-553–556, 2004.