

ユーザの技術に合わせた自動編曲機能をもつ ピアノ演奏練習システム

福田 翼¹ 池宮 由楽² 糸山 克寿² 吉井 和佳²
¹京都大学 工学部 情報学科 ²京都大学 大学院情報学研究科 知能情報学専攻

1. はじめに

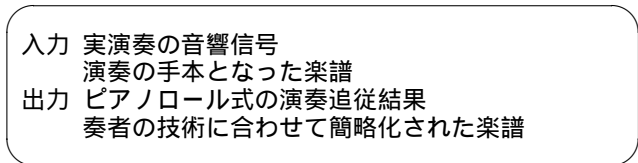
楽器の中でもピアノは特に演奏人口が多く、奏者の演奏レベルも非常に幅広い。ある程度演奏ができる中級者は自分の好きな楽曲で練習する傾向がある。これは更なる上達へのモチベーションとなる一方、そのような楽譜は演奏難度が高いことが多いため、意欲の低下を招くという問題がしばしば発生する。演奏難度の高い楽譜を奏者の演奏レベルに合わせて簡略化することができれば、この事態を避け、より効率的に演奏練習を促進することができる。

矢澤ら [1] は、ユーザが参照するギターの音響信号から、ユーザの演奏レベルに合わせた難易度の楽譜を生成するシステムを開発した。このシステムでは、ギター演奏の運指や押弦といった一般的な演奏の難しさを数値化することで、楽譜全体を編集している。

本稿ではピアノの演奏練習を対象とし、演奏楽曲の楽譜とユーザによる実際の演奏データ（音響信号）を入力とした、奏者の技術に合わせたピアノ演奏練習システムを提案する。本システムでは、和音分離により演奏データから生成した各音高ごとのアクティベーションを、参照楽譜と同期させることで時間的なずれを除去したピアノロールを生成する。何度かの演奏を経て奏者が苦手と思われるパートを判断し、該当箇所の簡略化を行うことで、楽譜を奏者の技術に合わせて編曲する。これにより、ユーザの得手不得手に細かく対応した演奏練習を実現できる。

2. ピアノ演奏練習システム

図 1 に本システムの概要図を示す。システムの入出力は以下の通りである。



2.1 音響信号の和音分離

非負値行列因子分解 (NMF) を用いて、ユーザ演奏の音響信号の和音分離を行う。入力音響信号の振幅スペクトログラム $V \in \mathbb{R}^{f \times n}$ が、あらかじめ用意した A0 から C8 までの 88 音の基底スペクトル行列 $W \in \mathbb{R}^{f \times 88}$ と、それらのアクティベーションを表す行列 $H \in \mathbb{R}^{88 \times n}$ という、2 つの非負値行列の積でモデル化できる ($V \approx WH$) と仮定のもと分離を行う。本稿ではピアノのみの演奏を対象としているため、先行研究 [2][3] からピアノ音の分離精度が最も良い β -ダイバージェンス NMF ($\beta = 0.6$) を用いる。

$$H \leftarrow H \otimes \frac{W^T((V \otimes WH)^{\beta-2})}{W^T(WH)^{\beta-1}}$$

ここで、 \otimes は要素ごとの積を表し、べき乗は要素ごとのべき乗を表す。また、基底スペクトル行列 W は、実際の演奏に使用する電子ピアノから各音高の音を録音し、一音ずつ基底数を 1 とした NMF により得られる基底 w_1, w_2, \dots, w_{88} の集合とする。

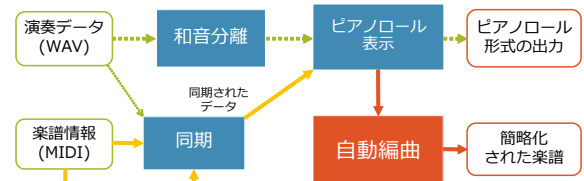


図 1 システム概要図

$$W = [w_1 w_2 \dots w_{88}]$$

これは、単音の音響信号に対して基底数 1 の NMF を用いさせることがその信号を正規化する、すなわち、その単音に対する基底ベクトルが出力として得られるという知見に基づいている。

NMF で得られたアクティベーションをしきい値処理により二値化することで、各音符の発音区間を検出する。これはユーザ演奏の音響信号をピアノロールで表現したものとみなすことができる。

2.2 楽譜と実演奏との同期

楽譜を実演奏と同期し時間的なずれを除去するため、動的時間伸縮法 (DTW) を用いる。DTW は、長さの異なる 2 つの時系列データを時間的に伸縮させることによって、それらの類似度を測るアルゴリズムであり、その過程で 2 つのデータの時間的な対応関係を得ることができる。まず、入力楽譜を MIDI データで表現する。このデータから TiMidity++[§] を用いて音響信号を生成し、短時間フーリエ変換を用いて、時間ごとのオクターブを無視した各音高のパワーを表す 12 次元のクロマベクトル u を計算する。ユーザ演奏の音響信号からも同様の方法でクロマベクトル v を計算する。これらを DTW により同期する。本稿では DTW に用いるベクトル間の距離として、コサイン類似度 $\cos(u, v)$ を採用した。

$$\cos(u, v) = \frac{\sum_i u_i \cdot v_i}{\sqrt{\sum_i u_i^2} \cdot \sqrt{\sum_i v_i^2}}$$

これによって u と v の時間的な対応付けを表すデータが得られる。

次に、楽譜の MIDI データから、全音符の音高・開始時間・継続時間を抽出し、上記で得られた時間的な対応付けを表すデータを元に音符の情報を変更することで、実際の演奏と時間的に同期の取れた楽譜データを作成する。

2.3 楽譜と演奏のピアノロール表示

和音分離で得られたユーザ演奏のピアノロールと、同期により得られた入力楽譜のピアノロールを比較表示する (図 2)。図の例では同期した楽譜情報を赤く、実演奏を和音分離したものを黒く表示しており、2 箇所 (丸に囲まれている箇所) が弾き間違いがあったことが分かる。

本研究では、奏者の苦手パートを判断するために弾き間違いの箇所を用いるが、NMF による和音分離では倍ピッチ/半ピッチ誤りが発生してしまうため、弾き間違いをしていないにもかかわらず弾き間違いと判断されてしまう問題がある。実際に図 2 においても、C7 や C8 周辺などの、楽譜には存在せず、奏者も演奏していない音高が、オクターブの弾き間違いのような形で採譜結果に現れている。

A Piano Tutoring System based on Automatic Music Arrangement: Tsubasa Fukuda, Yukara Ikemiya, Katsutoshi Itoyama, Kazuyoshi Yoshii (Kyoto Univ.)

[§] <http://twysynth.sourceforge.jp/>

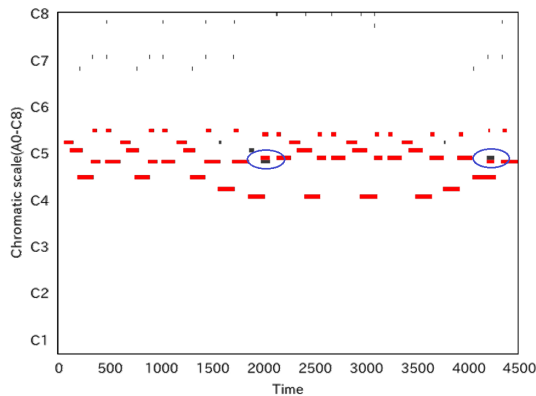


図2 ピアノロールの表示例

る．ソロのピアノ演奏を対象とした本研究では，認識率を大きく損なう原因である倍ピッチ/半ピッチ誤りが比較的少ない．そのため，ユーザが弾き間違えた箇所をより高い精度で判別することができる．ある時刻において弾き間違いと判定された音について，同時刻にオクターブ違いの音が楽譜通りに演奏されていた場合には，それを弾き間違いとはしないことにする．

2.4 奏者の技術に合わせた自動編曲

採譜によって弾き間違いが検出された周辺の箇所について，楽譜をパターン分けすることで簡略化を行う．具体的には以下に挙げる4つのパターンに分類し，各パターンごとにルールを与えることで自動的に編曲を行う．

2.4.1 パターン1：同時に弾く音数が多い楽譜

同時に弾く音数が多いパターンの楽譜では，和音の中で音高ごとに優先順位を設定し，それが低いものから取り除くことによって簡略化を行う．具体的には，まず和音内の最も高い音は基本的にメロディラインに含まれているため音楽的に最も重要な音であり，次いで和音のコードを構成する根音が重要な音となる．和音の中でこの2音は特に重要な音で，よほどの理由がない限り省略してはならないということが過去の自動編曲に関する研究 [4] でも述べられている．優先順位は根音の3度上の音，5度上の音と続いていくが，これらはいずれも上記で述べた2音ほど重要な音ではないため必要に応じて省略することができる．このような優先度を設定することによって，演奏の難しい和音を簡単な2和音，3和音に落とし込む．このパターンの簡略化の例を図3に示す．

2.4.2 パターン2：速い指の動きを要求する楽譜

速い指の動きを要求するパターンの楽譜については，一定以上のしきい値を越える速さの連続する音符をなくすことで簡略化を行う．この時，拍に乗っている音をできるだけ残すようにすることで，簡略化した後も音楽として違和感のないようにする．また，進行中のコードが楽譜データから分かるのであれば，コード内の音の優先度に応じて音を残すという手法も考えられる．このパターンの簡略化の例を図4に示す．

2.4.3 パターン3：離れた鍵盤を弾く楽譜

離れた鍵盤を弾く，いわゆる「跳躍」と呼ばれるパターンの楽譜については，さらに2通りの場合分けができる．すなわち，

- A 跳躍後すぐに跳躍前に演奏していた辺りに戻る場合
- B 跳躍後にしばらく跳躍がない場合

と分類することができる．Aのように離れた鍵盤を1つだけ弾かせるような場合には，その音を単純に削除し，Bのように離れた音符の近くにしばらく音符があるような場合には，跳躍の周辺の音を簡略化することで対処する．この際の簡略化のルールは基本的にパターン1とパターン2の方法に従う．A，Bのパターンの簡略化の例をそれぞれ図5，6に示す．

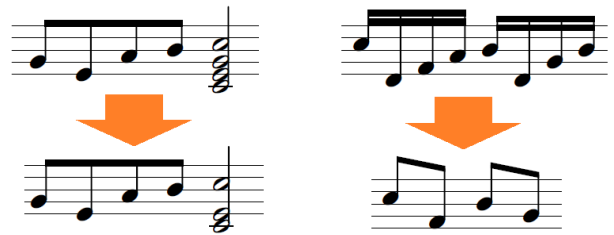


図3 パターン1の例

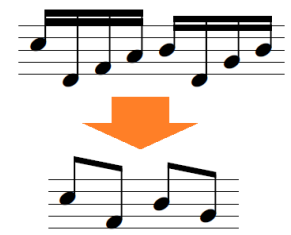


図4 パターン2の例

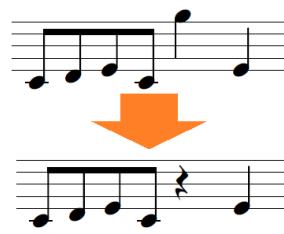


図5 パターン3Aの例

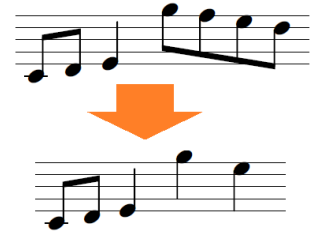


図6 パターン3Bの例

2.4.4 パターン4：上記以外の楽譜

ほとんどのミスは上記3パターンによってカバーできると考えられるが，このどれにも当てはまらないようなものも存在し得る．このような場合には，パターン1やパターン2の手法をしきい値を低く設定した上で適用することで対応する．すなわち，2和音を単音にしたり，4分音符以上の速さの音を削除したりすることによって簡略化を行うことができる．

3. 今後の課題

3.1 自動採譜の精度向上

NMFで得られたアクティベーションから倍ピッチ/半ピッチ誤りを取り除くことにより，ある程度の精度の向上を確認できた．しかしなおその精度は十分ではない．NMFの基底や分離結果の二値化の手法を改良することにより，更なる精度の向上が期待できる．

3.2 自動編曲の評価

編曲の結果が実際に音楽として成立しているかどうかは人によって感じ方が異なる．そこで，ピアノをある程度弾ける人に実際に演奏してもらい，それによって得られた楽譜を同じ人に演奏してもらうことで，生成された楽譜は元の曲と比べて違和感がないか，本生に自分の苦手な部分が簡略化されているかなどといった意見を反映することができる．このために複数人に対して被験者実験を行い，アンケートを取るなどして自動編曲の評価を行う予定である．

4. おわりに

本稿では，ユーザの技術や得手不得手に合わせて楽譜を簡略化する自動編曲を取り入れることで，より効率的にピアノ演奏練習を可能とするシステムを提案した．今後は自動編曲のルールの妥当性を検証するとともに，システム自体の改良も行っていく．なお，本研究の一部は，科研費 No.24220006(S), 26700020(A), 24700168(B) および OngaCREST プロジェクトの支援を受けた．

参考文献

- [1] 矢澤 他. ギター演奏者の習熟度に合わせた音響信号からのタブ譜自動生成. 情処研報, 2013-MUS-100(17):1-6, 2013.
- [2] E. Benetos *et al.* Score-informed transcription for automatic piano tutoring. In *EUSIPCO 2012*, 2012.
- [3] A. Dessen *et al.* Real-time polyphonic music transcription with non-negative matrix factorization and beta-divergence. In *ISMIR 2010*, pages 489-494, 2010.
- [4] G. Hori *et al.* Input-output HMM applied to automatic arrangement for guitars. *JIP*, 21(2):264-271, 2013.