

演奏しやすさを考慮した初心者向けドラム譜の自動採譜

Automatic drums score transcription for beginners by considering easy performance

曾我 蓉司郎
Yojiro Soga

法政大学情報科学部デジタルメディア学科
E-mail: 11k1025@stu.hosei.ac.jp

Abstract

In this paper, the tempo and bars are estimated but the input is a music sound and a drum sound signal. In addition, drum patterns are recognized that onset information is added. Drum patterns are repaired simply for beginners. Therefore, drum patterns are replaced simple patterns by the template matching that templates are assumed basic drum patterns. In evaluation experiments, music sounds are prepared for five and made matching to measure precision of the template matching. In result, precision of the template matching was about 97%. Precision of the template matching for replacing simply that is inputted a drum sound signal was 66%, and of the 8beat was 100%, 4beat was 88%, 16beat was 100% and back beat was 25%. Precision of the template matching that is inputted a music sound signal was 44%. From a result, template matching is effective. Drum scores are replaced simply without a sense of incongruity and this system is not be able to performance exercise support for beginners.

1 まえがき

楽器初心者は楽曲を練習する際に、楽譜を必要とする。しかし、楽曲によっては初心者では演奏が難しい物がある。世間には、初心者向けに簡略化されている楽譜が存在するが、数が少なく、楽曲も限られている。このような状況は初心者の楽曲演奏の妨げになる。自動採譜とは、音楽の音響信号波形から人間の代わりにコンピュータを使って楽譜を書き起こさせるプロセスである。楽曲の自動採譜は、ギターやピアノといった調波楽器のものが多く研究されている [1]。また、ドラムの様な打楽器は、教師なしクラスタリングを用いた手法や [2]、音源分離 [3] の研究がなされているが、ドラム譜を初心者向けに簡略化し、自動採譜する研究は行われていない。

そこで、楽曲の音響信号から、初心者でも演奏できるドラム譜へ自動変換し、採譜するシステムを提案する。楽曲のドラムパートは、8ビートの様なドラムパターンと16分音符等が連続するフィルインの二つに大きく分けられる。本稿では、ドラムパターンのみの簡略化を対象とする。ドラムセットは、膜鳴楽器（主に、スネア、バス、タムの三種類）、体鳴楽器（主にハイハット、リム、シンバルの三種類）、その他の打楽器から構成される。ドラムセットによってはスネアが複数使われ、タムが10種類以上存在する。体鳴楽器も同様に、シンバルが複数存在する。本稿では、ドラム3点と呼ばれる、スネアドラム（以下スネア）、バスドラム（以下バス）、ハイハットシンバル（以下ハイハット）で構成され

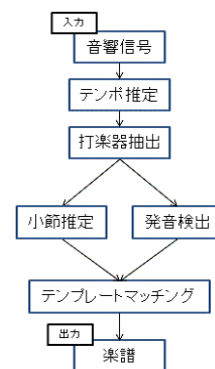


図1. システムの処理手順

るドラムパターンに簡略化する。対象とする楽曲は、J-POP や J-ROCK と呼ばれるジャンルとし、テンポは一定であり、変拍子は含まないものとする。

2 ドラム譜の自動採譜

2.1 システム概要

図1は、システムの処理手順を表す。まず、テンポ推定を行う。その後、楽曲を打楽器とそれ以外に分離し、小節推定と打楽器の発音検出を行う。小節毎に打楽器の発音部分についてテンプレートマッチングを行い、簡略化したドラム譜を採譜する。また、演奏者が演奏のイメージを持ちやすくするために、出力した楽譜のドラムパターンを再生できる。これにより、演奏者を聴覚的にも補助する。

2.2 ドラム譜の簡略化

楽譜を簡略化するに当たって、任意に改変すると、楽曲に合わなくなってしまうことがある。それを回避するために、楽曲に影響のない範囲で簡略化する必要がある。その対応策として、なるべく変えない点を考慮する必要がある。一点は、8ビートの部分は8ビートへ変換する等、ビートである。もう一点は、スネア等、耳に残りやすい音である。本稿では、これらをなるべく変えないことにより、楽曲に影響の無い範囲での簡略化を実現する。簡略化の方法としては、テンプレートとなるドラムパターンを用意し、テンプレートマッチングを行い、最も近いテンプレートに変換する。ここで用意するテンプレートパターンは、ドラムの教則本^{*1,*2}から初心者が演奏できる基本的なパターンを抜粋する。具体的には、8ビートが6個、4ビートが2個、16ビートが1個、裏打ちが2個、計11個用意した。これらのパターンで、初心者が練習しやすいJ-POP や J-ROCK といったジャンルの楽

*1山下浩, “これから始める!!ドラム入門” 株式会社ドレミ楽譜出版社,2007

*2山本雄一, “はじめてのドラム” 株式会社リトームミュージック,2011



図 2. 簡略化前 (上) 後 (下) の楽譜

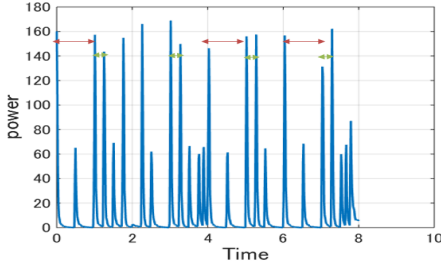


図 3. ドラム音源のパワーとビート間隔

曲をカバーする。

図 2 は上が簡略化前, 下が簡略化後の楽譜である。上の楽譜で, 先ほどの二点を考慮する。まず, これは 8 ビートのパターンになっているため, 8 ビートへ変換する。次に, 8 ビート内のスネアは耳に残りやすいため, この部分は変換しない。また, 小節の最後に 1 拍分フィルインが入っているが, この様に短いフィルインに関しては, ドラムパターンに変換してしまってもあまり影響は無い。これらの点について, 用意したテンプレートの中で最も近いものに変換した結果が, 下の楽譜になる。バスの音数が減っていることが特徴的であるが, J-POP や J-ROCK では, バスは多少減ってしまってもあまり問題はない。

2.3 テンポ推定

採譜するために, 小節を推定することが必須である。その際に, テンポを利用するため, テンポ推定が必要不可欠になる。

推定方法としては, 入力したドラムの時間波形について, 30 ミリ秒のフレーム毎にパワーを求め, その自己相関を計算し, 15 ミリ秒ずつシフトさせる。その自己相関を用いて音量変化の周期を検出する。ビート間隔の候補を推定し, テンポの候補となる値を出力する。テンポの候補はいくつか出力し, それぞれ出現回数も数える。出現回数が最も多いテンポ候補の中から, システム利用者に入力のテンポに近いものを選択してもらう。また, テンポ候補の BPM をメトロノームで再生することで, 楽器初心者でも最も近いテンポを選択できる。これにより, 誤ったテンポを検出して修正可能になる。

この手法での実験は, テンポが BPM165 と 108 の楽曲の推定を行った。BPM165 の楽曲では, 出現回数の最も多いテンポ候補は 83, 111, 167 で, BPM108 の場合は, 54, 87, 108, 143 が候補に挙げられた。多少誤差を生じているが, 実際のテンポに近い値が推定されている。

2.4 打楽器抽出

楽曲を入力としてテンプレートマッチングを行うため, 打楽器と調波楽器を分離させ, 打楽器のみを抽出する [4]。調波楽器は基本周波数とその倍音を含み, 打楽器は周期性のある音である。これらの特徴から, 調波楽器は周波数方向, 打楽器は時間方向に正常である。それぞれの成分をスペクトログラムの水平, 垂直な成分と見なすことで分離を行う。具体的には, 楽曲のスペクトログラムから各方向にグラフをスライスした配列を生成し, その配列に対して, メディアンを取ることで平滑化する。長さ 1 の信号 $x(n)$ を入力とした時の出力 $y(n)$ は

$$y(n) = \text{median}\{x(n-k : n+k), k = (l-1)/2\} \quad (1)$$

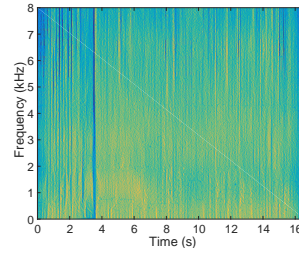


図 4. 分離した調波楽器

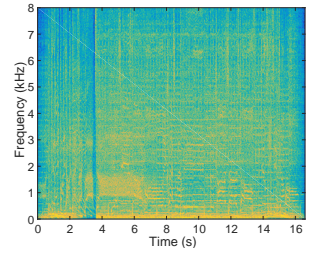


図 5. 分離した打楽器

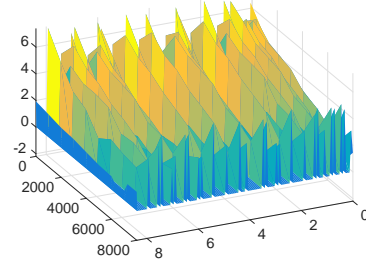


図 6. 発音検出例

で表す。(1) 式のメディアンフィルタを各方向に分割したスペクトルに対し適用することで, 調波・打楽器成分をそれぞれ強調する。ここで, S_i , S_h , M はそれぞれ i 番目の周波数方向のスペクトル, h 番目の時間方向のスペクトル, メディアンフィルタ処理である。

$$P_i = M\{S_i\} \quad (2)$$

$$H_i = M\{S_h\} \quad (3)$$

最後に, ウィナーフィルタを用いて各楽器成分を推定する。ウィナーフィルタとは, ノイズが可算される前の信号を推定するフィルタのことである。ウィナーフィルタは次のように設計される。

$$WF = \frac{H}{H+P} \quad (4)$$

2.5 発音検出

打楽器抽出を行った音源は, 打楽器のみが綺麗に抽出できているわけではない。そのため, このままテンプレートマッチングを行うと, 打楽器が発音していない部分で距離が大きくなり, 誤変換する場合がある。そこで, 発音検出を行い, 発音箇所のみに対してテンプレートマッチングを行う。

打楽器の発音検出の方法は, スペクトログラムをフレーム, 周波数帯域に分割し, ピークとなる帯域の総数が最も多いフレームを発音箇所とする [6]。ドラムパターン音源について, 発音検出の結果は図 6 となった。スネアとバスの発音検出を行い, 再現率が 0.62, 適合率が 0.79 であった。自動採譜の従来研究では, 打楽器の識別率が 90% 前後であり, それに比べるとあまり良い値ではない。

2.6 テンプレートマッチング

テンプレートマッチングを行うにあたって, マッチングさせる特徴量を決める必要がある。まず, ドラム 3 点についてスペクトル包絡を求めた。図 7~9 は, 赤がスペクトル包絡, 青がスペクトル周波数を表している。スネアは, 全体に緩やかな変化が見られる。バスは, スネアに比べて傾きが急な包絡である。ハイハットに関しては, スネアやバスとは違い, 高周波の成分を含んでいた。この様に, 3 点のスペクトル包絡に違いがはっきりと出ているため, この特徴量を用いる。

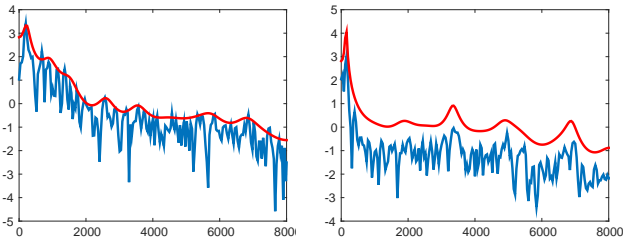


図 7. スペクトル包絡：スネア 図 8. スペクトル包絡：バス

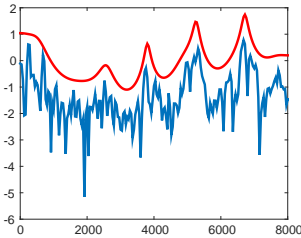


図 9. スペクトル包絡：ハイハット

具体的な方法は、用意したテンプレートと、入力したドラムパターンのスペクトル包絡の距離をフレーム毎にユークリッド距離で求める。そして、その距離が最も近いテンプレートに置き換える。また、テンポやドラムパターン、フレーム数が異なる場合に対応するため、DP マッチングを行う。図 10 の二つの楽譜のパターンでマッチングを行った例を挙げる。テンポやパターンが異なると、 i 番目のフレーム同士をマッチングさせると、全く違う部分同士をマッチングさせてしまう。また、マッチングされないフレームが現れる。これらを解決するために、DP マッチングを用い、最も近いと考えられるフレーム同士を対応させる様に伸縮し、最短距離を求める。図 10 の赤く囲った部分が二つのパターンで、唯一異なる部分になる。この部分に関して、 i 番目のフレーム同士をマッチングさせると、最短距離が求まらない場合がある。図 11 は、DP を用いたマッチング経路を表している。図 10 の赤く囲った部分に対応している経路は、図 11 の青く囲った部分である。この様に、最適なフレームをマッチングさせることにより、最も近いパターンを判別する。また、DP マッチングを行った際、局所的な伸縮を避けるために傾斜制限を用いる。図 12 は、本稿で用いた傾斜制限を表している。この傾斜制限を用いることで、縦横は 1 フレームずつしか伸縮できず、なるべく近いフレーム同士をマッチングさせ、最短距離を求めることができる。

3 評価

3.1 簡略化の妥当性

正解のテンプレートを決めるために、簡略化の妥当性を測る実験を行った [7]。簡略化したいドラムパターンとすべてのテンプレートを楽器経験のある被験者 5 人に聞かせ、変換しても違和感無いと感じるテンプレートを選択させる。最も多く選択されたテンプレートを正解テンプレートとする。しかし、選択された数が過半数に至らなかった場合、最も数の多いテンプレートすべてを正解テンプレートとする。

3.2 テンプレートマッチングの精度

まず、ドラム音源を用意し、テンプレートマッチングを行い、精度を測った。FFT の点数は 1024 点、フレーム幅 1024 とし、256 点ずつシフトさせた。スペクトル包絡は、LPC を用い、フィルター次数 21 で求めた。入力はテンプレートをランダムに並べたドラム音源 (以下音源 1,2) を、それぞれ 16 小節用意した。テンポはそれぞれ BPM120,140 にして、GrageBand で打ち

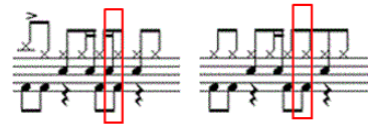


図 10. マッチング例

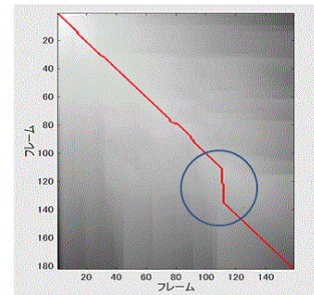


図 11. マッチング経路

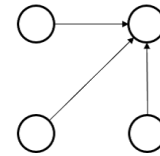


図 12. 用いた傾斜制限

込んだ。テンプレートは 1 小節の物を 8 ビートが 6 個、4 ビートが 2 個、16 ビートが 1 個、裏打ちが 2 個、BPM はすべて 120 の物を合計 11 個用意した。同じテンプレートに変換されると正解とし、正解になった小節数でテンプレート正解率を求めた。また、正解テンプレートに変換されない場合、何ビートに変換されたかを判断する、ビート正解率も求めた。

テンポ推定の結果、音源 1 は BPM121 となり、音源 2 は BPM140 となった。推定結果を用いて、小節毎に分割し、テンプレートマッチングを行った。音源 1 のテンプレート正解率は約 93%、ビート正解率はすべて 100% になった。音源 2 のテンプレート正解率、ビート正解率は共に 100% になった。このことから、テンプレートマッチングが有効であることが示せた。

3.3 簡略化の精度

簡略化の精度を測った。3.2 と同様の条件にし、実際の楽譜から簡略化したい部分を 4 小節ずつ抜粋し 4 つ繋いだドラム音源 (以下音源 3,4) を、それぞれ 16 小節として用意した。テンポも同様に、BPM120 と 140 とし、GarageBand で打ち込んだ。また、3.1 で述べた様に、小節毎に正解テンプレートを決め、テンプレート正解率とビート正解率を求めた。ビート正解率を調べることで、変換しにくいビートを把握できる。

テンポ推定の結果は 3.2 と同様の結果となった。音源 3,4 のテンプレート正解率はそれぞれ約 69%、約 62% になり、ビート正解率は、表 1 に示す。

表 1. 音源 3,4 のビート正解率

ビート	音源 3	音源 4
8 ビート	100%	100%
4 ビート	100%	75%
16 ビート	100%	100%
裏打ち	25%	25%

図 13,14 はそれぞれ左の小節が音源 3 から抜粋した 1 小節で、右の小節が正解のテンプレートである。図 13 は、正しく変換できたが、図 14 は正解でないテンプレートに変換された。図 13 は、音源の楽譜と正解のテンプレートのスネアは一ヶ所異なる。それに対して図 14 の楽譜では、スネアの位置は一致しているが、バスの位置が一ヶ所しか一致してない上に、テンプレート



図 13. 簡略化対象と正解テンプレート



図 14. 簡略化対象と正解テンプレート



図 15. 誤変換されたテンプレート

にはバスは四ヶ所含まれている。図 15 は、図 14 の音源が誤変換されたテンプレートである。図 14 の音源とは、バスが一ヶ所とハイハットが異なる。したがって、ハイハットよりスネアやバスが一致していれば、距離が小さくなる。つまり、スネアやバスが余分に演奏されていると、距離は大きくなる。図 14 の音源では、テンプレートはバスが音源より多く演奏されていたため、正解との距離が大きくなり、誤変換された。また、音源 3 に対して、正確なテンポを与えた結果、テンプレート正解率は約 94% に上昇した。テンポが 1 でもずれると、テンプレート正解率が下がるため、より正確なテンポ推定が必要になってくる。また、入力とテンプレートのテンポが異なる場合でも、テンポが同じ場合に比べると精度が下がる。DP マッチングを行うだけでなく、テンプレート全体を伸縮させることで、改善される可能性がある。

3.4 楽曲を入力としたテンプレートマッチング

簡略化されている楽譜が存在する楽曲に対して、楽曲を 8 小節に切り取った音源を入力として二つ用意した (以下音源 5,6)。音源 5 のテンポは BPM165、音源 6 は BPM79 である。打楽器抽出を行い、同様の手順でテンプレートマッチングを行った。簡略化されている楽譜を参考にし、正解のテンプレートを決定した。

テンポ推定の結果は、音源 5 が BPM167、音源 6 が BPM80 になった。テンプレート正解率は音源 5 が 37.5%、音源 6 が 50%、全体で 44% になった。発音検出した箇所のみでテンプレートマッチングを行うと、距離を計算する部分が限られてしまい、テンプレート毎の差が小さくなる点が欠点である。発音箇所が少ないテンプレートでは、自然と距離が小さくなり、そのテンプレートに変換される可能性が高くなる。入力を楽曲にした場合、楽曲から打楽器音を抽出して行うため、音が劣化してしまい、ドラム音源を入力とした場合に比べて、マッチングの距離は自然と大きくなる。また、打楽器抽出すると、スネアやバスは明白に認識できるが、ハイハットの音が聞こえにくいので、ハイハットの発音検出ができない。したがって、楽曲から正確な発音検出を行うことで、改善される可能性がある。

4 あとがき

4.1 結論

本稿では、テンポ推定、打楽器抽出、小節推定を行った。また、打楽器の発音検出、テンプレートマッチングを用いたドラム譜の簡略化、及び採譜を行った。テンプレートをランダムに並べた音源に対してテンプレートマッチングを行った結果、テンポ推定が正確な場合、テンプレート正解率は 100%、テンポ推定

が誤った結果になると、テンプレート正解率は 93% になった。16 小節のドラム音源に対してドラム譜の簡略化を行い、テンプレート正解率は全体で約 66% であった。ビート正解率は 8 ビートが 100%、4 ビートが 50%、16 ビートが 75%、裏打ちが 0% という結果になった。また、楽曲を入力とした場合、テンプレート正解率は全体で 44% になった。このことから、テンプレートマッチングは有効である。また、簡略化に関しては、一部のドラムパターンで有効であるが、全体の精度は低く、改善の余地がある。

4.2 今後の展望

今後の展望としては、ビート正解率が最も低かった裏打ちに関して、テンプレートマッチングの場面で重み関数を使用し、バスの値を低く、スネアとハイハットの値を高くすることでビート正解率を上昇させる。これは他のビートに関しても、それぞれのビートの特徴となる部分に重み関数を使用することで、簡略化を可能にする。また、DP マッチングの際に、傾斜制限にペナルティを用いることで正解とは大きく違うテンプレートを変換対象から外せる。比較している二つの信号が似ていて、テンポが異なる場合、DP マッチングで縦のみや横のみが多く含まれる。しかし、二つの信号が大きく異なっている場合、縦と横共に多く含まれる場合がある。これは、二つの信号の似ているフレームが全然違う箇所が存在することが予想される。つまり、DP マッチングで縦と横にペナルティをつけ、そのペナルティがある一定の値を超えると、たとえ距離が小さくても変換対象から外すことで、誤変換を避けることができる。また、ハイハット検出を行うことができれば、ビートを確定することが可能になる。ビート毎にテンプレートマッチングを行うことにより、簡略化の精度向上が見込める。今後、テンプレートの数を増やせば、簡略化の度合いを変化させて、あらゆるドラマーのレベルに対応できる。楽曲から抽出した打楽器音の発音検出とテンポ推定をより正確に行うことで、簡略化の精度を上昇させる。結果、楽曲を入力することが現実的になり、今まで以上に初心者への支援になる。

参考文献

- [1] 坂内, 他, “くし形フィルタに基づく自動採譜システムの実現” 情報処理学会研究報告, 2007(81), pp13-18
- [2] 吉井, 他, “自己認識マップによる教師なしクラスタリングを利用したドラム演奏の自動採譜” 情報処理学会研究報告, 2003(82), pp43-50
- [3] 後藤, 他, “打楽器を対象とした音源分離” 電子情報通信学会論文誌, D-II, 情報・システム, II-情報処理 J77-D-2(5), pp901-911, 1994
- [4] Derry FitsGerald, “HARMONIC/PERCUSSIVE SEPARATION USING MEDIAN FILTERING”, DAFX10, Graz, Austria, September 6-10, 2010.
- [5] 内田誠一, “DP マッチング概説～基本と様々な拡張～” 電子情報通信学会技術研究報告, PRMU, パターン認識・メディア理解 106(428), pp31-36, 2006
- [6] Yongwei Zhu, et.al, “DRUM LOOP PATTERN EXTRACTION FROM POLYPHONIC MUSIC AUDIO” ICME, 2009
- [7] 山崎, 他, “極座標表現を用いた形状特徴ベクトルによる 3 次元ビデオのセグメンテーション” 情報処理学会論文誌, コンピュータビジョンとイメージメディア, Vol47, pp208-217, 2006