

対話型進化計算法を使用した ドラムワンショット音源選択支援システム

清瀬 一志^{2,a)} 福嶋 政期¹ 荒川 豊¹

概要: Desk Top Music (DTM) とは音楽を制作する際に PC を使用して制作する手法である。DTM を行う際に各楽器パートを入力 (打ち込み) していくが、ドラムパートはワンショット音源という 1 度だけドラムを叩いた音を鳴らしたい時間に設置してドラムのパートを作成していく。また、ドラムパートはバスドラムやスネアドラム、ハイハットシンバルなど約 10 種類の個別のドラムで構成されている。他の楽器パートは入力段階で音色を変える手段を持っているが、ドラムパートには存在しない。そのため事前に数百数千種類のさまざまな音色のワンショット音源を保存しておき、作成する曲の雰囲気に近い音源を事前に選択する必要がある。この時求めているものを見つけるまでに時間がかかってしまう、後から聞き直した時この音源が最適ではなかったなどの問題が存在している。これは保存しているワンショット音源が膨大であり最適なものを早くに見つけられないからである。この課題を解決する方法として、対話型進化計算法を使用して保存しているワンショット音源から作曲者が対象曲に最適だと感じる音源を選択させるソフトウェアを作成する。本論文では提案システムに最適な選択肢の数と特徴量の情報提示によるユーザー疲労の差を実験によって明らかにする。

1. はじめに

近年、音楽を制作する方法は多様化しており、特にパソコンで音楽を制作する手法を Desk Top Music (DTM) と呼び、多くの人がこの手法を行なっている。また、DTM の中にもすでに録音や作成した音源を特定の時間に配置する手法や楽器を接続して録音を行う手法、楽器の機能を搭載したプラグインというソフトウェアで擬似的に演奏を行う手法など多様である。すでに録音や作成した音源を特定の時間に配置する手法において、ドラムパートでは 1~4 小節録音した音源 (ループ音源) をループさせる手法と 1 度だけドラムを叩いた音源 (ワンショット音源) を自分の鳴らしたい時間に設置することでドラムパートを構成する手法が存在している。ワンショット音源は手間がかかる、基礎知識がないと難しいというデメリットが存在するが、自分が作りたいと思っている構成を作ることができるためこの手法が採用される場面が多い。この時、ワンショット音源自体が作曲者の求めている音と違う、今制作している曲調と合わないといった場合、プラグインや外部ソフトウェアで音を編集するという手法が取られる。しかし、この手法は既存のワンショット音源からエフェクトをかけていくため

編集できる幅が限られており、エフェクトをかけても満足する音を得られなかった場合はワンショット音源自体を変更する必要がある。そのため、この手法でドラムパートを制作する作曲者は 1 つのドラムあたり数百という多数のワンショット音源を保存している。また、ドラムパートにはバスドラムやスネアドラム、ハイハットシンバルなど 10 程度のドラムが存在しているので、作曲者が所有するワンショット音源の総数は数千程度になることもある。

そのため、作曲者の所有している音源を一括管理するソフトウェア^{*1}が存在するが、このシステムではすべての音源が一覧できるようになっており、選択肢が多すぎて選択するまでに手間がかかってしまうという問題が存在する。

そこで、本研究では対話型進化計算法を使用し、多数存在するワンショット音源の中から作曲者がその時求めている音を選択させるシステムの提案を行う。

本稿ではまず、ドラムの音色に着目した先行研究と対話型進化計算法を使用したシステムに関する先行研究について紹介をする。その後、ドラムのワンショット音源の音響特性を独自に分析した結果に基づき要件を設定し、その要件を満たすための対話型進化計算法を用いたワンショット音源選択支援システムの提案及び構築方法について述べる。最後に、提案システムのユーザビリティを評価するために

¹ 九州大学大学院システム情報科学研究院

² 九州大学大学院システム情報科学府

^{a)} kiyose.kazushi.844@s.kyushu-u.ac.jp

^{*1} COSMOS, <https://wavesjapan.jp/plugins/cosmos>

2つの実験を行った。第一実験では選択肢の数を要因とし、選択肢の数によって判断に要する時間や認知負荷が変わることを確認した。第二実験では特徴量空間の表示をする実験を行い、特徴量空間を表示することは逆に認知負荷がかかりユーザーの疲労が軽減されないことを確認した。

2. 関連研究

本研究はドラムパートのワンショット音源の選択支援を行うシステムを提案するものなのでまずドラムパートの音色に関する研究について紹介を行い、その後対話型進化計算法を音楽分野に使用した研究についての紹介を行う。

2.1 ドラムパートに音色に関する研究

ドラムパートの音色に関する研究は主に音色の情報提示 [3] と音源生成 [8][9]、クラスタリング [2][4][16] が行われている。

Lee ら [3] はドラムのワンショット音源の音響特性に大きな影響を与える情報を決定し、これらの情報を選択時に視認できるシステムを提案している。しかし、情報提示のみ行なっているので認知負荷が上がるのが考えられる。そのため、音源選択の支援やユーザビリティなどにより選択までの試行回数を減らす仕組みが必要となる。

音源生成 [8][9] では敵対的生成ネットワーク (GAN) を使用して、作曲者の求めている音を生成する手法である。この手法のメリットとして、作曲者の所有するワンショット音源が少なくても良い、何度も試行することで求めている音色のワンショット音源を生成する可能性が高いという2点が存在する。

クラスタリング [2][4][16] では周波数解析を行い、音響特性の情報を使用しドラムパートの分類を行っている。しかしこれらの研究では主にスネアドラムとバスドラム、ハイハットシンバルなど別々のドラムを分類するものであり同じドラムを対象としている本研究とは意義が異なる。

以上のことをふまえ、今回のシステムでは同じドラムのワンショット音源を選択する際に作曲者が実際に聞いて選択を行う、その選択の試行回数を少なくするシステム設計が必要となる。

2.2 対話型進化計算法に関する研究

対話型進化計算法は、進化計算の分野の中で人間が直感的な評価を行うことによって解を求めることができる最適化手法である。ユーザーは与えられた選択肢の中で直感的に自身の好みに合う選択肢を選ぶことでユーザーの嗜好に近い選択肢が提示されていく。この手法は定量的な評価を出すことが難しい、人の好みや主観に依存する問題に有効とされている。そのため音楽、特に楽曲推薦 [11][13][15] や音色 [7]、作曲 [1][10] の分野でよく使用されている。音色 [7] の分野ではアナログシンセサイザーなど既にパラメータが

存在しており、そのパラメータの数が多い場合に有効である。作曲の中でも特定の楽器に対して対話型進化計算法を用いた研究としてはギター [12] やドラムパート [5][6][14] が存在する。現状、ドラムパートの取り組み [5][6][14] は全て対話型進化計算法を使用しリズムパターンをループ音源形式で生成させるものである。ドラムパート音源の音色を選択する際に対話型進化計算法を取り入れた研究は調べた限りでは存在しない。また、対話型進化計算法には評価の試行回数が多い (世代を経る) ことによる疲労、また疲労による評価精度の低下などの問題が存在する。そのため、人間が評価しやすいようなシステム設計を行う必要がある。

3. 提案システム

本章では対話型進化計算法を取り入れたワンショット音源選択支援システムについて説明を行う。提案システムでは対話型進化計算法において音響特性の近い音源を選択させるため、各音響特性を特徴路用とした特徴量空間が必要となる。そのため、まずドラムパートのワンショット音源の音響特性の情報から特徴量を取り出し、特徴量空間を作成する方法を説明し、その後特徴量空間を使用した音源選択支援システムについて説明を行う。

3.1 音響特性の情報と特徴量空間の作成

Lee ら [3] がドラムパートのワンショット音源の音響特性に大きな影響を与えるとした5つの情報について説明を行う。

3.1.1 Length (長さ)

ここでの長さとはワンショット音源自体の長さではなく、1度ドラムを叩いた時の音の持続時間のことを指す。例としてスネアドラムにはスナッピーというパーツがついており、このスナッピーの種類によって残響音が変わる。

3.1.2 Peak Energy Frequency (ピークエネルギー周波数)

ピークエネルギー周波数とは一般的には音の高さを指す。それぞれのドラムにはチューナーと呼ばれる締め具がついており、締め具合によってピークエネルギー周波数が変わる。また楽曲の年代やジャンルによってドラムのピークエネルギー周波数が変化しており、重要な要素の1つである。

3.1.3 Overtone (倍音)

倍音とは周波数の中でピークエネルギー周波数以外の周波数成分である。ドラムには非整数次倍音が多く含まれるとされているが、同じ状態の楽器でも叩き方などによって倍音の成分は異なってくる。

3.1.4 Tapping sound (タッピング音)

タッピング音とはドラムを叩いた時の接触音とされており、基本的にピークエネルギー周波数よりも高い周波数成分で構成される。また、タッピング音は瞬間的に消えるとされている。

3.1.5 Transient shape (トランジェント)

トランジェントとは波形が最大になるまでの間に発生する音を指す。よりパーカッシブな音ほどトランジェントが大事とされており、ドラムにおいては必要不可欠な要素である。

3.1.6 特徴量空間の作成

上記 5 個の音響特性のうち Length, Peak Energy Frequency (PEF), Overtone の 3 個の情報は波形データやスペクトログラムを処理することにより、数値化することができる。これらの処理を保存しているワンショット音源すべてに対して行い、1 つの空間に散布図のように配置することで 3 次元の特徴量空間を作成することができる。

3.2 対話型進化計算法を取り入れた提案システム

今回の提案システムの流れを図 1 に示し、実際にユーザーに表示するインターフェースを図 2 に示す。

- (1) 特徴量空間全体の中からランダムに複数個のワンショット音源が選択肢とする。
- (2) 選ばれた選択肢をユーザーに提示する。
- (3) ユーザーが音源を 1 個選択する。ここで、ユーザーが選択する音源はユーザーが制作している楽曲の雰囲気に近いものや想像している音色の音源である。
- (4) ユーザーの選んだ音源をもとにユーザーの好みの座標を変更する。選択回数 n の時、式 (1) のような座標変更を行う。この座標変更によりそれまでの選択の情報が残された座標を作ることができる。

$$\left(\frac{\sum_{i=1}^n i \cdot \alpha_i}{\sum_{i=1}^n i}, \frac{\sum_{i=1}^n i \cdot \beta_i}{\sum_{i=1}^n i}, \frac{\sum_{i=1}^n i \cdot \gamma_i}{\sum_{i=1}^n i} \right) \quad (1)$$

- (5) 好みの中心座標 $(\alpha_1, \beta_1, \gamma_1)$ とそれぞれの音源座標 $(\alpha_2, \beta_2, \gamma_2)$ のユークリッド距離 d を式 (2) のようにそれぞれ算出する。

$$d = \sqrt{(\alpha_2 - \alpha_1)^2 + (\beta_2 - \beta_1)^2 + (\gamma_2 - \gamma_1)^2} \quad (2)$$

- (6) 距離の近いものから順に特定の個数を選択肢の候補とする。この個数は世代を経るごとに特定の減少率で候補個体を減らしている。
- (7) 選択肢候補の中からランダムに選択肢の数から 1 個減らした音源数を次の選択肢とし、選択肢候補外から 1 個ランダムに選び次の選択肢とする。選択肢候補外も次の選択肢に含めているのは局所最適から脱却するためである。
- (8) (2) から (7) を繰り返し、ユーザーにとって最適な音源が出た場合は確定してシステムを終了する。

4. 実験

本章では対話型進化計算法を取り入れたワンショット音源選択支援システムの評価実験について説明を行う。まず、ドラムのワンショット音源を使用した際の最適な選択肢の

数を明らかにするため、表示する選択肢の数を比較した実験を行い (実験 1)、その結果をもとに特徴量の情報提示によりユーザーの疲労が軽減されるか明らかにするため、特徴量空間の提示による比較実験を行う (実験 2)。

4.1 使用するデータ

今回の 2 つの実験ではシステムの評価にあたり、splice^{*2}にアップロードされているスネアドラムのワンショット音源 400 個を使用する。使用する音源をモノラルに変換した後、特徴量空間の作成のために以下の操作を行なった。

Length: 図 3 の緑の間の範囲で示すように振幅が最大値の 1/20 になる位置を最大値より前と後に見つけその間の長さを特徴量としている。

PEF: 図 4 で示すように Length の範囲の間で高速フーリエ変換 (FFT) を実行し、得られたスペクトルを 1Hz ごとに再サンプリングした後 50Hz から 1500Hz の範囲で振幅が最大となる周波数を特徴量としている。

Overtone: PEF の周波数からその 5 倍までの周波数について、1Hz ごとに最大振幅との比を算出し合計した数値を特徴量としている。

また、それぞれの特徴量が正規化されていないので最小値を 0、最大値を 1 とした正規化を行っている。

今回の音源データから算出した特徴量空間は図 5 となる。

4.2 実験 1: 表示する選択肢の数に関する実験

対話型進化計算法を使用したシステムにおいて選択肢の数が増えることはユーザーの選択肢を増やすので、ユーザーへの負担が増加する。しかし、選択肢を減らすとユーザーにとっての最適な音源が見つかるまでの時間が増えることや最適な音源が見つからないまま終了してしまうことにつながる。そのため最適な選択肢の数を本実験において調査する。

4.2.1 実験手順

本実験では選択肢の数は 3 個から 6 個までの 4 種類を用意した。また、この選択肢のうち 1 個は前回選んだ選択肢である。実験参加者は図 6 の目指す音源のボタンに近い音源をまず音源 1 から 4 と前回選んだボタン聞き、その横の選択肢ボタンを押し選んでいく。ここで、もし目指す音源と同じ音だと感じたらさらにその横の確定ボタンを押す。本実験では定量的なデータ取得のため、ユーザーが目指す音源を設定し、その音源に近い音源を選ばせるようにした。また、選択回数の最大値を 10 回とし、それまでに最適な音源が見つからない場合は 10 回目に選択肢した音源を最適な音源とした。音源数の関係から選択肢候補決定の際の減少率は 2/3 とした。音源自体の長さで判断することを防ぐた

*2 splice, <https://splice.com>

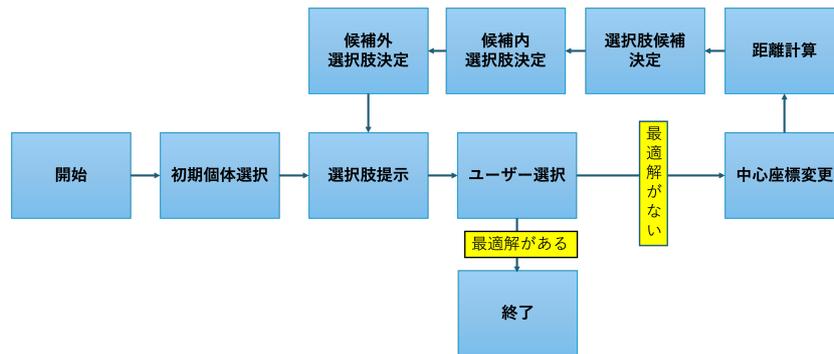


図 1: 提案システムの流れ



図 2: 提案システムのインターフェース左の音源ボタンで音源を聞き, 真ん中の選ぶボタンでユーザーの求めている音に近い音源を選択する. 求めている音が出たら左の確定ボタンを押す.

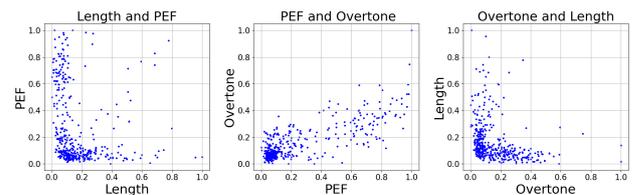


図 5: 使用するデータの特徴量空間

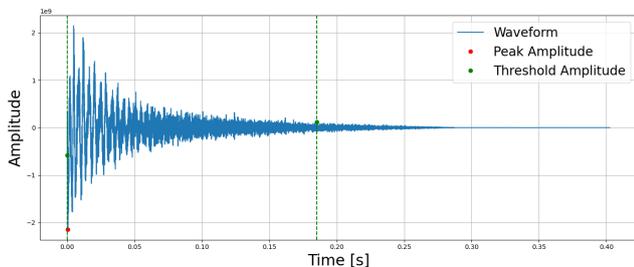


図 3: スネアドラムの波形
 (緑の線で囲まれた範囲が Length の長さとなる)

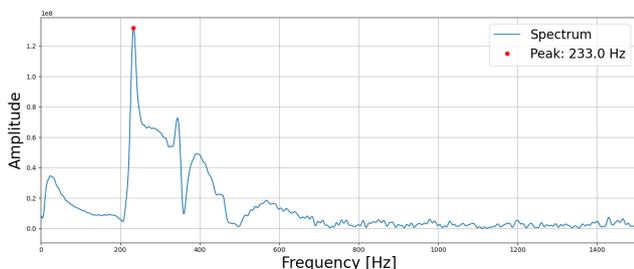


図 4: スネアドラムのスペクトラム
 (赤い点が Peak Energy Frequency の周波数となる)

め選択肢の音源を聞く際の最大時間を 1.5 秒に設けた. これは特徴量空間の Length の最大値が 1.37 秒だったため



図 6: 選択肢の数を変更する実験システムのインターフェース

(左の音源ボタンで音源を聞き, 目指す音源のボタンに近い音源を真ん中の選ぶボタンで近い音源を選択する. 目指す音源と同じ音源だと感じたら右の確定ボタンを押す.)

の範囲をカバーできる秒数として設定した. また, 音量差で判別することを防ぐために, 音量の平滑化を行なった.

この実験では音楽演奏の経験者であり, DTM の経験はない 5 人を実験対象とし, それぞれの選択肢の数の実験を 2 回ずつ計 8 回行なった.

4.2.2 実験結果

結果を表 1 に示す. 1 世代で音源を聞いた回数は選択肢が増えるのと同時に増えている. システムが終了するまでの総時間と目標音源と最終選択の音源の距離は選択肢が 4 個と 5 個の間に大きな差が見え, 5 個と 6 個の時の差はあ

まり見られなかった. このことから, ワンショット音源を聞き選択する際の最適な選択肢は 5 個の場合に精度が上が
 り疲労が少ないと確認できた.

表 1: 選択肢の数をパラメータとした場合のユーザビリティと精度の関係

選択肢の数	1 世代で聞いた回数	総時間	距離
3 個	3.82 回	53.60 秒	0.66
4 個	5.24 回	62.91 秒	0.63
5 個	6.58 回	82.94 秒	0.17
6 個	7.90 回	81.13 秒	0.14

4.3 実験 2: ユーザーに特徴量空間を見せる実験

ユーザーが音源の選択を行う際に, 今回の選択肢や過去の特徴量空間の推移などを表示することで聞かなくて良い選択肢や近い選択肢など知り, 選択回数やシステム終了までの総時間が減り, 認知負荷が減ることが考えられる.

4.3.1 実験手順

本実験では先述の実験から選択肢の数を 5 個としている. 音源の選択方法や変数などは先述の実験と同じである. この実験では選択肢に色が割り振られており, その色をした音源が特徴量空間上でどこに存在するのか右側の図から見るができるようになっている. 今回は 3 次元の特徴量で実験を行なっているため, 見やすさの観点から 2 つの特徴量を選択するボタンを選択ボタンの下に設けている. また, Matplotlib のナビゲーションツールバー機能を使用し, 拡大や移動によって見やすさを確保している.

この実験では音楽演奏の経験者であり, DTM の経験はない 6 人を実験対象とし, 特徴量空間の表示がある時とない時の実験を 5 回ずつ計 10 回行なった.

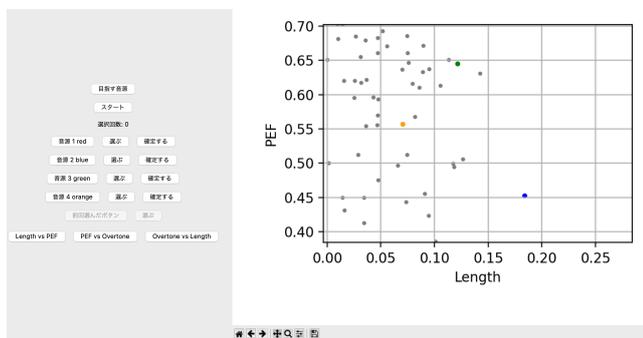


図 7: 特徴量空間を見ることができる実験システムのインターフェース

(左の音源ボタンに色に対応した音源が同じ色で右の特徴量空間に表示されている. 左下のボタンを選択することで特徴量空間の x 座標 y 座標を変更できる. 下のボタンを操作することにより特徴量空間内で拡大や移動を行うことができる.)

4.3.2 実験結果

結果を表 2, 各世代の聞いた回数と選択までの時間の推移を図 8 に示す. この結果から横に特徴量空間の表示がある場合とない場合での音源の聞く回数や結果に大きな差はないこと, システムの終了までの総時間が増えることがわかった. また世代を経ても特徴量空間の表示がある場合はない場合よりも選択までの時間が長くなることがわかった. この理由としては, 一度図を見て判断する時間がかかる, 世代を経て図を拡大するのに時間がかかるなどの理由が考えられる.

実験参加者からの意見として特徴量空間内で目指す音源から遠そうな音源でも目指す音源の可能性はあると押ししてしまう, Length や PEF は直感的にわかるが Overtone の特徴量は分かりづらいので判断に使用しなかったなどが挙げられた.

表 2: 特徴量空間の表示のユーザビリティと精度の関係

特徴量空間	1 世代で聞いた回数	総時間	距離
表示あり	6.40 回	140.81 秒	0.15
表示なし	6.38 回	103.16 秒	0.17

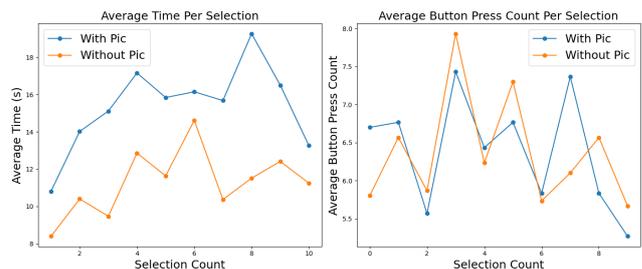


図 8: 世代ごとの聞いた回数, 時間の推移

5. 考察

今回の実験においてシステム終了までに対象音源が見つからなかった実験の全体図を図 9 に, 拡大図を図 10 に, 世代ごとの選んだ音源と対象音源との距離を図 11 に示す. Length の特徴量においてより対象音源に近い音源を選ぼうとした結果 Overtone の特徴量が離れてしまっている. また, 5 世代以降に対象音源から離れてしまっているため, 近い音源が選択されずに距離が離れたまま終了してしまった. そのため, ユーザーが対象音源と遠いと感じたら対象音源から候補内選択肢の数を広げる, 候補外選択肢を増やすなどのシステム設計が必要である.

また, 特徴量的には遠い座標にある音源でもユーザーの判断ではほとんど変わりがないといったことや, 逆に特徴量的に近いと感じてもユーザーの判断で遠い座標だったという事例が存在した. この原因としては Tapping Sound や Transient Shape の今回特徴量としていないことや, スネア

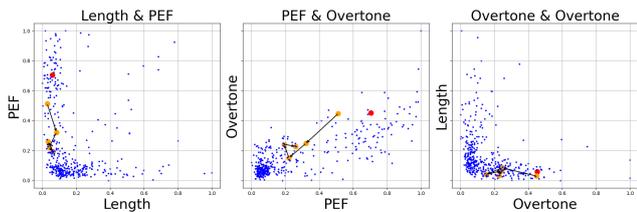


図 9: 特徴量空間でのユーザーの選んだ音源の推移
 (赤い点が目標音源で黄色い点が選択した音源とその推移)

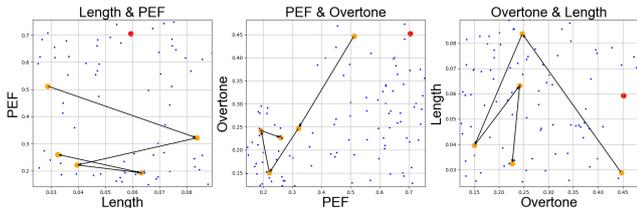


図 10: 図 9 の拡大図



図 11: ユーザーの選んだ選択肢と対象音源との距離
 (青い点が選んだ音源の推移を表して赤い点が選ばれなかった選択肢を示す)

ドラムのスナッピーの金属音が反映され PEF が本来の音よりも高く計測されたことなどが理由として考えられる。

6. まとめ

本研究はユーザーの作曲をより円滑に行うことを目的とし、作曲者がワンショット音源を選ぶまでに手間がかかってしまう問題を解決するため、少ない試行回数でユーザーにとって最適なワンショット音源を選択するためのドラムワンショット音源選択支援システムを作りその評価実験を行なった。対話型進化計算法を使用したシステムにおいてどのようなシステム設計でユーザーの疲労が軽減できるかを調査した結果選択ボタンは 5 個の場合に疲労が少なく、精度が高くなることがわかった。また、現在特徴量空間を表示することは逆に認知負荷がかかりユーザーの疲労が軽減されることがわかった。

今後の課題としては、特徴量としていない Tapping Sound と Transient Shape の数値化、世代を経た後に対象音源との距離がある場合において選択肢の幅を広げるようなシステム設計があげられる。また、本研究では特徴量によりユー

ザーの好みの範囲を収束させたが、特徴量の差がユーザーに感知できないほど小さい場合、ユーザーの嗜好に合うと判断して良い。そのため、特徴量空間において差異を区別できなくなる距離の傾向調査などの定性的なデータ取得が必要である。

謝辞

本研究は、科学研究費補助金 (JP23H00216) の支援のもと実施されている。

参考文献

- [1] Biles, J.: GenJam in Perspective: A Tentative Taxonomy for GA Music and Art Systems, *Leonardo*, Vol. 36, pp. 43–45 (online), DOI: 10.1162/002409403321152293 (2003).
- [2] Chhabra, A., Veer Singh, A., Srivastava, R. and Mittal, V.: Drum Instrument Classification Using Machine Learning, *2020 2nd International Conference on Advances in Computing, Communication Control and Networking (ICACCCN)*, pp. 217–221 (online), DOI: 10.1109/ICACCCN51052.2020.9362963 (2020).
- [3] C.H.Lee, S.: Exploration of Practical Information Entry Technique of One-Shot Drum Audio Samples, *Smart Technology Applications*, Vol. 1, No. 1, pp. 21–28 (2020).
- [4] Herrera, P., Yeterian, A. and Gouyon, F.: Automatic Classification of Drum Sounds: A Comparison of Feature Selection Methods and Classification Techniques, *Music and Artificial Intelligence* (Anagnostopoulou, C., Ferrand, M. and Smaill, A., eds.), Berlin, Heidelberg, Springer Berlin Heidelberg, pp. 69–80 (2002).
- [5] Hoover, A., Rosario, M. and Stanley, K.: Scaffolding for Interactively Evolving Novel Drum Tracks for Existing Songs, pp. 412–422 (online), DOI: 10.1007/978-3-540-78761-7_44 (2008).
- [6] Kaliakatsos-Papakostas, M. A., Floros, A. and Vrahatis, M. N.: evoDrummer: deriving rhythmic patterns through interactive genetic algorithms, *Proceedings of the Second International Conference on Evolutionary and Biologically Inspired Music, Sound, Art and Design, EvoMUSART'13*, Berlin, Heidelberg, Springer-Verlag, p. 25–36 (online), DOI: 10.1007/978-3-642-36955-1_3 (2013).
- [7] Luke, S.: *Stochastic Synthesizer Patch Exploration in Edisyn*, pp. 188–200 (online), DOI: 10.1007/978-3-030-16667-0_13 (2019).
- [8] Nistal, J., Lattner, S. and Richard, G.: DrumGAN: Synthesis of Drum Sounds With Timbral Feature Conditioning Using Generative Adversarial Networks (2022).
- [9] Rouard, S. and Hadjeres, G.: CRASH: Raw Audio Score-based Generative Modeling for Controllable High-resolution Drum Sound Synthesis (2021).
- [10] Tzimeas, D. and Mangina, E. E.: A GA Tool for Computer Assisted Music Composition, *International Conference on Mathematics and Computing*, (online), available from (<https://api.semanticscholar.org/CorpusID:10303749>) (2007).
- [11] Unehara, M., Yamada, K. and Shimada, T.: Subjective evaluation of music with brain wave analysis for interactive music composition by IEC, *2014 Joint 7th International Conference on Soft Computing and Intelligent*

Systems (SCIS) and 15th International Symposium on Advanced Intelligent Systems (ISIS), pp. 66–70 (online), DOI: 10.1109/SCIS-ISIS.2014.7044758 (2014).

- [12] Walker, T. M. and Whalen, S. P.: Dynamic Recombination of Evolving Guitar Sounds (DREGS): A Genetic Algorithm Approach to Guitar Synthesizer Control, *2013 IEEE International Symposium on Multimedia*, pp. 248–254 (online), DOI: 10.1109/ISM.2013.47 (2013).
- [13] Yamaguchi, G. and Fukumoto, M.: A Music Recommendation System based on Melody Creation by Interactive GA, *2019 20th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD)*, pp. 286–290 (online), DOI: 10.1109/SNPD.2019.8935654 (2019).
- [14] Yamamoto, R., Nakashima, S., Ogawa, S. and Fukumoto, M.: Proposal for Automated Creation of Drum’s Fill-In Pattern Using Interactive Genetic Algorithm, *2011 International Conference on Biometrics and Kansei Engineering*, pp. 238–241 (online), DOI: 10.1109/ICBAKE.2011.37 (2011).
- [15] 斉藤優理, 伊藤貴之: MusiCube : 特徴量空間における対話型進化計算を用いた楽曲提示インタフェース, 可視化情報学会論文集, Vol. 34, No. 9, pp. 17–27 (オンライン), DOI: 10.3154/tvsj.34.17 (2014).
- [16] 中平 徹, 田中敏幸: 周波数解析によるドラム音のパターン分類, パーソナルコンピュータ利用技術学会論文誌, Vol. 4, No. 1, pp. 15–20 (オンライン), DOI: 10.24663/jjpcats.4.1.15 (2010).