

キー音を用いたリズムアクションゲームにおける 譜面の自動生成

福永 大輝^{1,a)} 越智 景子^{2,b)} 大淵 康成^{2,c)}

概要:「リズムアクションゲーム」とは、楽曲のリズムに合わせた「譜面」と呼ばれる視覚的な記号に沿って操作を行うゲームである。また楽曲中の音のうち、操作に応じて発音されるものを「キー音」と呼ぶ。譜面には、各再生音に対するユーザー操作の情報などが記述されている。本研究の目的は、キー音を用いたリズムアクションゲームの制作を容易にするために、任意の楽曲から譜面を自動生成するシステムを開発することである。そのために本研究では、楽曲中の音の音響的特徴と再生される時系列の情報から機械学習を行い、譜面を自動生成する手法を提案する。提案手法により自動生成した譜面について、ユーザー評価を行った。ランダムに配置された譜面と比較して、自動生成した譜面の楽しさが向上していることが確認できた。一方で、適切な難易度のモデリングについては改善が必要であることがわかった。

キーワード: 音楽, 音響特徴, 機械学習, クラスタリング, リズムアクションゲーム, キー音

1. はじめに

楽曲のリズムに合わせた「譜面」と呼ばれる視覚的な記号に沿ってプレイヤーが操作を行い、その正確さによって得点が記録されるゲームを「リズムアクションゲーム」と呼ぶ [1]。演奏する記号一つを「ノート」と呼び、譜面はノートの集合である。譜面は一般的にイメージされる音楽の楽譜のようなものではなく、リズムアクションゲームのゲームデータに相当する。

リズムアクションゲームにおいて、プレイヤーが行った操作に応じて再生されるサウンドは「タップ音」と「キー音」の2つに分類することができる。タップ音とは楽曲とは関係のない効果音であり、本来の楽曲の音とは別に重ねて再生されるものである。また、キー音とは楽曲中の個々の音を抽出して切り出されたものである。譜面通りの正確な操作が行われない際、キー音を用いたゲームでのみ楽曲自体がインタラクティブに変化する。そのため、タップ音を用いたゲームよりもキー音を用いたゲームは高いインタラクティブ性を持つと考えられる。

本研究の最終的な目的は、任意の波形データの楽曲から、キー音を用いたリズムアクションゲームの譜面を自動生成することである。譜面を作成するにあたっては、以下のような過程が必要とされる。

- (1) 楽曲から個々の音を抽出し切り出す
- (2) 個々の音についてキー音とBGMの分類を行う
- (3) デバイスに合わせキー音を譜面上に適切に配置する

ここでBGMとは、プレイヤーの操作にかかわらず自動的に再生される音である。楽曲中の全ての個々の音は、BGMかキー音のいずれかに割り当てられる。

第1過程は、完全な自動化がなされているわけではないが、補助を目的としたツールが多数存在している。第2過程、第3過程は自動化がなされておらず、質は譜面作成者の経験に依存するため、良い譜面の作成には熟練を要する。本研究では、既に第1過程を終えたデータを対象として、第2過程と第3過程について扱う。

我々はこれまで、楽曲中の個々の音を持つ音響的特徴と、それらの音の発音時点の情報を使用し、第2過程について機械学習による自動的な分類手法を提案した [2]。さらに、学習データをクラスタリングすることにより譜面を持つ個性をモデリングし、分類精度を向上させた [3]。

そこで本研究では、第3過程についても機械学習による手法を適用し、実際に譜面の自動生成を行った。また、ゲームとしての楽しさは分類精度のみからでは判断できないため、生成した譜面についてユーザー評価を行った。

¹ 東京工科大学大学院バイオ・情報メディア研究科
メディアサイエンス専攻
Graduate School of Bionics, Computer and Media Science
Tokyo University of Technology

² 東京工科大学メディア学部
School of Media Science Tokyo University of Technology

a) g311801818@edu.teu.ac.jp

b) ochikk@stf.teu.ac.jp

c) obuchiysnr@stf.teu.ac.jp

2. 関連研究

Donahue らは、楽曲の特徴に相当するスペクトル情報とニューラルネットワークを利用して、自動的に譜面を生成する手法を提案している [4]。指定する難易度が低いほど品質が下がることが課題であり、精度改善が望まれている。プレイヤーの操作による発音が存在しないリズムアクションゲームである StepMania を学習データとしており、キー音を用いたゲームのケースは想定されていない。

キー音を用いたリズムアクションゲームにおける譜面自動生成については、香川らによって MIDI 形式の楽曲に対して自動的に譜面データを生成する手法が提案されている [5]。リズムアクションゲームに用いられる楽曲は、MIDI 形式ではなく WAV 形式などの波形データであることが多い。そのため、手法を適用可能な楽曲が限定されてしまうという問題がある。

同様に、キー音を用いたゲームについて、Lin らによって波形データである楽曲を入力とした手法が提案されている [6]。切り出された楽曲中の音を楽器別にラベリングし、4 層の全結合層を持つモデルによって譜面の自動生成を行っている。あらかじめ用意された楽器のラベルに含まれない音が存在し得ることや、同じ楽器の音でも大きく音響的特徴が異なる場合があることについては考慮されていない。また、難易度の低い譜面と高い譜面を区別せずに学習することによって、分類精度が低下している可能性があることが示唆されている。これは、難易度をはじめとした譜面が持つ特性によって、譜面作成における分類基準が異なるためである。

3. 使用データ

PC 用のキー音を用いたリズムアクションゲームである BMS[7] のデータを使用した。BMS のプレイ画面を図 1 に示す。7 つの鍵盤と丸いスクラッチのレーンが存在し、ノートはいずれかのレーンに配置される。

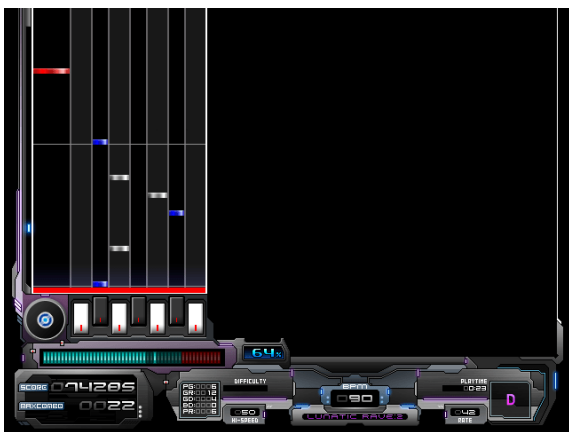


図 1 BMS のプレイ画面

Fig. 1 The play screen of BMS.

BMS データは、楽曲中の個々の音が切り出された多数のサウンドファイルと、難易度が異なる 1 個～数個の譜面ファイルから成る。譜面ファイルには楽曲中の個々の音の再生時点が記述されている。さらに個々の音がキー音か BGM か、キー音の場合はスクラッチ、またはいずれかの鍵盤を選択する記述がなされている。

譜面作成における第 2 過程には個々の音をキー音、または BGM に分類するタスクが存在し、第 3 過程にはキー音を鍵盤とスクラッチに分類するタスクと、鍵盤に分類された音を 7 レーンのいずれかに割り当てるタスクが含まれる。

譜面中のノートはキー音である個々の音に相当する。ノートが多いほどプレイヤーに要求する操作は多くなるため、本研究では楽曲の長さから 1 秒あたりのノート数、つまりノートの密度を算出し、譜面難易度の指標とした。

本研究では、熟練したプレイヤーによって BMS データが 399 譜面を含む 85 曲分選定されたパッケージである、GENOSIDE -BMS StarterPackage-[8] を使用した。

4. 特微量

譜面を自動生成するにあたり、楽曲が持つ特徴を学習することを目的として、データセットに含まれる全てのサウンドファイルについて、音響特微量の抽出を行った。

音響特微量の抽出には、openSMILE[9] を使用した。特微量を抽出するにあたって、全てのサウンドファイルを 16bit/44.1kHz/2ch に変換する前処理を行った。特徴抽出は 25ms のフレームサイズ、10ms のフレームシフトでフレーム特微量 (LLD) を抽出し、その後各フレーム特微量に対して全フレーム分の特徴を表す全体特微量 (Functional) を抽出する方式で行った。

音響特微量の数は、フレーム特微量の数と全体特微量の数を乗算した数である。本研究ではフレーム特微量を 6 種類、全体特微量を 10 種類使用したため、60 種類の音響特微量を使用した。使用したフレーム特微量の一覧を表 1 に、全体特微量の一覧を表 2 に示す。

譜面ファイルからは、再生時点順に並んだ全ての個々の音を持つ情報が得られる。全ての個々の音について、同時または直前の時点に存在する 5 個の音の探索を行う。それらの音を持つ音響特微量と各タスクにおけるラベル、分類する音との時点差の情報についても使用する。

表 1 使用したフレーム特微量 (LLD)

Table 1 The list of used LLDs.

名称	説明
LogEnergy	音量 (対数)
HNR	調波成分と非調波成分の比率
F0	基本周波数
SpectralFlux	スペクトルの変化量
SpectralVariance	スペクトルの分散
SpectralSkewness	スペクトルの歪度

表 2 使用した全体特徴量 (Functional)
Table 2 The list of used Functionals.

名称	説明
Max	最大値
Min	最小値
Range	最大値と最小値の差
LinRegC1	線形回帰係数 (傾き)
LinRegC2	線形回帰係数 (オフセット)
LinRegErrQ	線形回帰二乗誤差
Reg-centroid	回帰直線の重心
Skewness	歪度
Kurtosis	尖度
MeanPeakDist	ピーク間の距離の平均

フレーム特徴量数を L , 全体特徴量数を F , 探索する個々の音の数を P としたとき, 機械学習に用いる特徴量の次元 D は (1) で表すことができる. 本研究では, 371 次元の特徴量を用いた.

$$D = P(LF + 2) + LF + 1 \quad (1)$$

各タスクにおける機械学習によるラベルの分類には, Weka[10] を用いた. また学習アルゴリズムには, サポートベクターマシン (SVM) の最適化手法の一つである, 逐次最小問題最適化法 (SMO) [11] を用いた.

5. クラスタリング

譜面には, 各タスクの分類基準によってさまざまな個性が存在することが示唆されている [12]. そのため, この個性をモデリングするためにクラスタリングを行った.

まず, データセットに含まれる譜面の全ての対について相互に学習・評価を行い, 2 つの分類の正解率の平均を譜面間の類似度とする. この類似度に基づき, (2) に定義されるクラスタリングを行う. s_i は i 番目の譜面を, C_k は k 番目のクラスターを表す.

$$x_{ik} = \begin{cases} 1 & (s_i \in C_k) \\ 0 & (\text{else}) \end{cases} \quad (2)$$

コスト関数 C を (3) に定義する. G_{ij} は譜面 i と j の類似度である. Metis[13] を用いて最小カット問題を解くことで, C を最小化するクラスタリングが得られる.

$$C = \sum_{ijk} G_{ij}(1 - x_{ik}x_{jk}) \quad (3)$$

クラスタリングにより, データセット中の譜面は複数のクラスターに分類される. このとき, 各タスクにおけるラベルの分類基準が似た譜面が同じクラスターに属する. 分割するクラスターの数 10 で試行し, 無作為に 1 つのクラスターを選択した. 選択したクラスターに含まれる譜面を学習データとして, 機械学習のモデルを作成した.

6. ユーザー評価

提案手法によって生成した譜面がゲームとして楽しいものとなっているか, ユーザー評価を行うことで検証する. ユーザー評価に用いる楽曲には, 既存の低難易度の譜面が含まれていることと, よく知られている楽曲であることの 2 つの条件から, 「Stand by me」を使用した.

まず, 譜面作成における第 1 過程を終えたデータを入力として, 譜面作成における第 2 過程と第 3 過程の分類を機械学習により行い, 実際に譜面の自動生成を行うシステムを実装した. このシステムにより生成した譜面を「提案手法により生成した譜面」とする. また, 指定した難易度に応じて第 2 過程と第 3 過程の分類をランダムに行った譜面を用意した. これを「ランダムに配置した譜面」とする. さらに, 「Stand by me」の BMS データ作者が作成した譜面についても, 比較のため使用した. これを「人手で作成された譜面」とする.

被験者の多くはリズムアクションゲームに熟練していないと考えられることから, 人手で作成された譜面は難易度が最も低いものを選択した. ノートの密度の条件に差異がないように, ランダムに配置した譜面と提案手法により生成した譜面のパラメータを調整した.

ユーザー評価の被験者は 30 人 (10~30 代, うち男性 25 人, 女性 5 人) の大学生・大学院生である. 評価項目は譜面の楽しさ, 難しさ, 音楽と合っているかの 3 項目であり, 項目の選択肢は 5 段階評価とした. また, リズムアクションゲームのプレイ頻度によって結果に差が生まれる可能性があるため, 譜面の評価とは別にこれらについてもアンケートに含めた. 各譜面がどのように作られたかは伏せ, さらに被験者によってプレイする譜面の順番が異なるように配慮した. プレイ頻度別に被験者の回答を集計し, 5 段階評価の平均を得た. ユーザー評価の結果を図 2 に示す.

まず, 譜面の楽しさについては, ランダムに配置した譜面の評価が最も低く, 人手で作成された譜面と提案手法により生成した譜面の評価が同程度となった. また, プレイ頻度が低いグループでは人手で作成された譜面の評価が最も高かったが, プレイ頻度が高いグループでは提案手法により生成した譜面の評価が高かった.

次に, 譜面の難しさについては, いずれのグループでも, 人手で作成された譜面が最も易しく, ランダムに配置した譜面と提案手法により生成した譜面がほぼ同程度に難しいという結果であった. ノーツの密度に差がないのにも関わらず難易度に明らかな差が生まれた理由として, 人手で作成された譜面では譜面全体におおよそ均等にノーツが存在するが, それ以外の譜面では全くノーツが存在しない箇所とノーツが極端に多い箇所が存在することが挙げられる. つまり, ノーツの密度だけでは難易度の指標として不十分であることがわかる.

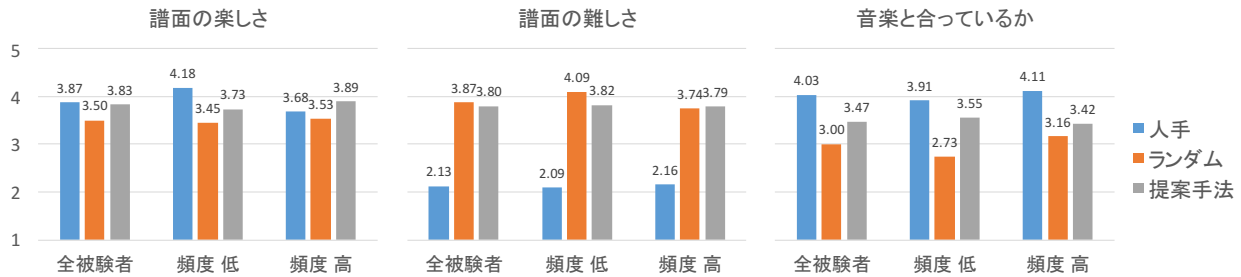


図 2 ユーザー評価の結果

Fig. 2 The result of user evaluation.

現在のシステムには、このようなノートの偏りを制御することができるパラメータが存在しない。リズムアクションゲームにおいて難易度が適切であることは非常に重要であるため、譜面の自動生成を行ううえで難易度の制御は大きな課題であるといえる。

最後に、譜面と音楽が合っているかについては、人手で作成された譜面が最も評価が高く、ランダムに配置した譜面が最も評価が低かった。提案手法により生成した譜面の評価はランダムに配置した譜面より高く、一定の範囲で音楽のリズムやメロディなど目立つ音をキー音として選択できていることが示唆される。プレイ頻度が低いグループと比較すると、プレイ頻度が高いグループでは提案手法により生成した譜面とランダムに配置した譜面の評価の差が小さかった。このことから、リズムアクションゲームをよくプレイする人ほど、リズムが完全に合っていない部分に対しての違和感が大きいのではないかと推測される。

また、全ての被験者の回答における譜面の楽しさと譜面と音楽が合っているかの項目について、譜面 1 で $r = 0.45(p < 0.01)$ 、譜面 2 で $r = 0.40(p < 0.05)$ 、譜面 3 で $r = 0.48(p < 0.01)$ となり、有意な正の相関が見られた。

7. まとめ・展望

キー音を用いたリズムアクションゲームを対象として、楽曲中の個々の音から得られる音響特徴量と、譜面から得られる過去の時系列情報を用いて機械学習を行うことで、指定するノートの密度に応じた譜面を自動生成するシステムを実装した。さらに、実際に譜面の自動生成を行った。

ユーザー評価の結果からは、自動生成した譜面が楽しいものとなっていることを一定の範囲で確かめられた。現在のシステムでは、難易度に関する情報がキー音の密度のみであり、時間的なキー音の偏りや同時押し配置を制御することができないことが課題として挙げられる。

将来的には、難易度のモデリングを適切に改善することや、熟練したプレイヤーにとっても違和感のない配置を実現することが望まれる。また、主成分分析などによる特徴量の次元削減や深層学習モデルの適用を行うことについても、譜面の質を向上させるうえで有効であると考えられる。

参考文献

- [1] Karen Collins: *Game Sound: An Introduction to the History, Theory, and Practice of Video Game Music and Sound Design*, pp. 74–75, The MIT Press (2008).
- [2] 福永大輝, 越智景子, 大淵康成: リズムアクションゲームにおけるキー音の自動推定, 芸術科学会論文誌, Vol. 18, No. 1, pp. 10–18 (2019).
- [3] Daiki Fukunaga, Keiko Ochi, Yasunari Obuchi: Training Data Clustering for Key-Sound Estimation in Rhythm Action Games, *NICOGRAPH International 2019*, pp. 70–73 (2019).
- [4] Chris Donahue, Zachary C. Lipton, Julian McAuley: Dance Dance Convolution, *Proceedings of the 34th International Conference on Machine Learning*, Vol. 70, pp. 1039–1048 (2017).
- [5] 香川俊宗, 手塚宏史, 稲葉真理: 音楽の重要な構成要素の抽出の提案-音楽ゲーム用譜面自動生成のために-, エンタテインメントコンピューティングシンポジウム 2015 論文集, pp. 326–333 (2015).
- [6] Zhiyu Lin, Mark Riedl, Kyle Xiao: GenerationMania: Learning to Semantically Choreograph, *Proceedings of the 15th AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, Vol. 15, pp. 51–57 (2019).
- [7] Urao Yane: BM98Data.format.specification, available from (<http://bm98.yaneu.com/bm98/bmsformat.html>) (accessed 2020-02-20).
- [8] lobsak: GENOSIDE -BMS StarterPackage-, available from (<http://nekokan.dyndns.info/~lobsak/genoside/>) (accessed 2020-02-20).
- [9] Florian Eyben, Felix Weninger, Florian Groß, Björn Schuller: Recent Developments in openSMILE, the Munich Open-Source Multimedia Feature Extractor, *Proceedings of the 21st ACM international conference on Multimedia*, pp. 835–838 (2013).
- [10] Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, Ian H. Witten: The WEKA Data Mining Software: An Update, *SIGKDD Explorations*, Vol. 11, No. 1, pp. 10–18 (2009).
- [11] J. C. Platt: Sequential Minimal Optimization: A Fast Algorithm for Training Support Vector Machines, Technical Report MSR-TR-98-14, Microsoft Research (1998).
- [12] 辻野雄大, 山西良典, 山下洋一, 井本桂右: ダンスゲーム譜面の特性分析とクラスタリングに基づく特徴的な譜面の自動生成, エンタテインメントコンピューティングシンポジウム 2019 論文集, pp. 96–103 (2019).
- [13] George Karypis, Vipin Kumar: A fast and high quality multilevel scheme for partitioning irregular graphs, *SIAM Journal on Scientific Computing*, Vol. 20, No. 1, pp. 359–392 (1998).