

リズムゲームにおける既存入力指示符パターン 混合による多様な譜面自動生成

小林樹生^{†1} 栗原渉^{†1} 三上浩司^{†1}

概要：リズムゲームでは同楽曲における多様な譜面の制作は難しい。本研究では、入力指示符の配置をパターンとして分類し、そのラベルを用いた機械学習と生成結果の加工により、意図的にパターンを混合した譜面を自動生成する手法を提案する。結果としてパターンを意図的に持たせることは可能で、配置変化も既存手法より表現できたが、ロング入力指示符を含む場合は特徴の反映が不十分であり、今後はさらなる分析を行うことを計画している。

1. はじめに

ゲームは多くの人に親しまれており、その一分野としてリズムゲームが存在する。リズムゲームとは、楽曲に合わせて決められたタイミングで視覚的に示される特定の入力操作を行い、その正確さを競うゲームのことである。このとき、入力のタイミングや内容は入力指示符によって表される。また、ある楽曲に対応する一連の入力指示符の集合は譜面と呼ばれる。譜面に対して入力するときの魅力は様々あるが、その1つとして、入力を行う際に手や指を動かす心地よさがある。プレイヤーはその動きを促す譜面の得手不得手にかかわらず、特定の動きに対して高揚感を覚え、繰り返し行いたいと考えることも多い。

しかし、譜面は一般的に、譜面制作専用ツールを用いてプロの譜面制作者によって制作されるため、遊べる譜面は用意されたものであり、必ずしもプレイヤーにとって心地よい譜面とは限らない。そのため、好きな曲をプレイできたとしても、用意された譜面に対応する動きが原因で、リズムゲームにおける継続的なプレイ意欲や満足度に悪影響を与える可能性がある。

この問題に対して、自分好みに譜面をカスタマイズすることができれば、プレイヤー個人が心地よいと感じる動きにパーソナライズ化できるという解決策が考えられる。しかしながら、自分で譜面を制作しようとした場合、譜面制作ツールが存在しなければ不可能である。また、存在していたとしても、音楽的知識およびリズムゲームの経験、ツールの知識といったスキルが必要である。さらに、ツールが存在し必要なスキルを有していた場合でも、譜面の制作には多くの時間がかかる。

この問題への解決策、あるいは譜面制作者の支援手法として、楽曲から譜面を自動生成する研究が行われている。しかし、これらの研究における自動生成手法は、難易度や譜面単位の特徴といった全体的指標に基づく制御を主としており、一定の時間間隔ごとの配置傾向を意図的に制御す

ることは困難である。特に、既存譜面に見られる、ある区間における複数の入力指示符が持つ配置の傾向（以下、「入力指示符パターン」と呼ぶ）を設計可能な手法は、現時点では十分に確立されておらず、プレイヤー個人へのパーソナライズ化は難しい。

そこで、本研究では、プレイヤーが意図した入力指示符パターンが反映された譜面を、楽曲から自動生成する手法を提案する。本手法では、楽曲と各入力指示符パターンの混合割合を入力として用いる。これにより、人手によって制作された譜面の傾向を保持しつつ、指定した混合割合を適切に反映した譜面の生成を目指す。

2. 関連研究

既存譜面のような傾向を持つ譜面およびその特徴を持つ譜面を自動で生成する研究はいくつか行われている。

Chris, D.ら[1]は、CNN 付き RNN を用いた機械学習を用いて、楽曲データと指定した難易度ラベル（Easy や Challenge といった譜面の難易度を示す指標）を入力として譜面を生成する手法を提案し、既存譜面に近い傾向を持つ譜面生成が可能であることを示した。

辻野ら[2]は、譜面から抽出した譜面特徴量と音響特徴量を用いてクラスタリングを行い、譜面数が多かった3つのクラスタについて、Chris, D.ら[1]と同様の手法によって機械学習を行うことで、特徴的な譜面を生成する手法を提案した。

福永ら[3]は、プレイヤーの入力と同時に楽曲の一部が再生されるリズムゲームを対象として、より詳細な音響特徴量を用いた譜面の自動生成手法を提案した。この研究では、譜面同士を相互に学習・評価して得られる正解率に基づく類似度を用いて譜面のクラスタリングを行った。

しかしながら、これらの手法はいずれも譜面単位での配置傾向の再現にとどまっており、同一譜面内に存在する入力指示符配置における一定の時間間隔ごとの傾向変化を考慮することはできない。そのため、区間ごとの特徴である

^{†1} 東京工科大学

入力指示符パターンは扱われていない。

一方で, Emily, H.ら[4]は, 入力指示符の出現時刻と種類を同時に推定するモデルを提案し, Chris, D.ら[1]の手法と比較して, 既存譜面に見られる入力指示符パターンをより高精度に再現できることを示した。しかし, この手法における入力情報は音楽情報のみに限られており, 意図した入力指示符パターンを譜面へ反映させることはできない。

したがって, 混合割合によって意図的に入力指示符パターンを設計可能とする点, および複数の入力指示符パターンを組み合わせた新たな入力指示符パターンも適用可能とする点が, 本研究の新規性である。

3. 入力指示符と入力指示符パターン

リズムゲームにおける入力のタイミングや内容は入力指示符によって表される。最も多くのリズムゲームに共通する入力指示符として, 通常入力指示符とロング入力指示符が存在する。各入力指示符の操作を図1に示す。

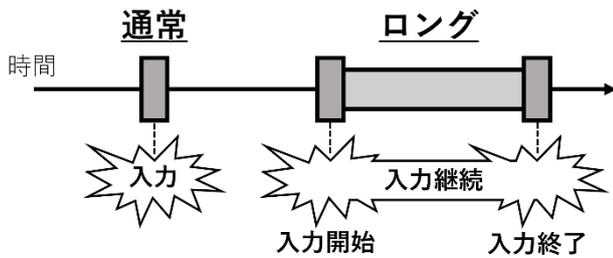


図1 通常入力指示符とロング入力指示符の操作

通常入力指示符は単一の入力タイミングを示し, ロング入力指示符は入力開始および入力終了タイミングの両方のタイミングを示す。

入力指示符パターンとは, 既存譜面に見られる, ある区間における複数の入力指示符が持つ傾向のことである。既存譜面では, 様々な入力指示符パターンの区間が組み合わせることで, 1つの譜面が構成されている。

本研究では, 一連の入力指示符に従い入力する際のプレイヤーの手の動きの傾向に着目する。このとき, 入力装置は一直線上に4つ並んでいると仮定する。また, プレイヤーの間では, 一方の手に負担が偏るのを避け, 同時にその負担もできるだけ最小限にするという慣例がある。これにならない, プレイヤーは必要のない限り左右合わせて3本以上の指を使用せず, 主に左右それぞれ1本の指で入力し, 可能な限り左右の手を交互に使用すると仮定する。

本研究における入力指示符パターンの例として, 同一レーンに連続して通常入力指示符が出現したり, 左右の手が交差するような配置の通常入力指示符が出現したりする区間およびその入力指示符群の配置が挙げられる。また, ロング入力指示符など, 異なる種類の入力指示符の存在によ

っても, 入力指示符パターンは変化する。入力指示符パターンの具体例を図2に示す。

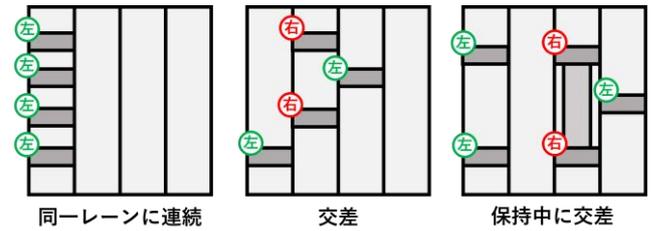


図2 入力指示符パターンの具体例

4. 提案手法

本章では, 本研究の提案手法について述べる。まず, 本手法の概要について述べた後, 重要な3つの処理についての詳細を述べる。

4.1 本手法の概要

本手法は, モデル学習および生成という2つの段階に分かれる。まず, モデル学習段階の処理を行い, その結果を用いて生成段階の処理を行う。

モデル学習段階の全体図を図3に示す。この段階の目的は, 意図した入力指示符パターンが反映された譜面を生成するための学習済みモデルを用意することである。まず, 4時刻4レーンにおける入力指示符の全配置を「分類処理」によってグループ分けする。分類結果に基づき, 既存譜面データの各時刻がどの入力指示符パターンに該当するのかを表す情報を付与する。最後に, 入力指示符パターンの情報が付与された譜面データを用いて「学習処理」を行い, 学習済みモデルのパラメータを保存する。保存したパラメータは, 生成段階で利用する。

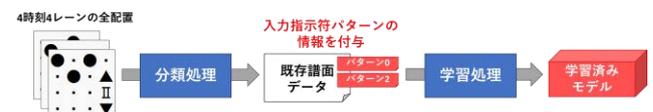


図3 モデル学習段階の全体図

生成段階の全体図を図4に示す。この段階の目的は, 入力された情報に従い譜面を生成することである。

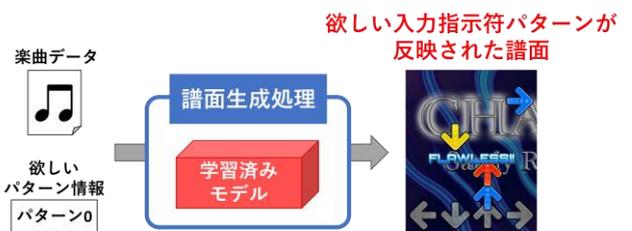


図4 生成段階の全体図

楽曲データと欲しい入力指示符パターンの混合割合を用いて「譜面生成処理」を行い、入力情報が反映された譜面を生成する。譜面生成処理では、モデル学習段階で保存したパラメータを読み込んだモデルを利用する。

4.2 分類処理

本研究では、4 時刻 4 レーンにおける入力指示符の配置を対象とし、プレイヤーの手の動きの観点から類似する配置同士をまとめる。

入力指示符は、通常入力指示符およびロング入力指示符の 2 種類のみを対象とする。ロング入力指示符については、開始・保持・終了といった状態の違いを区別せず、同一の入力指示符として扱う。これは、プレイヤーの手の動きが変化しない範囲でより単純化するためである。

プレイヤーの手の動きの推定では、プレイヤーは必要のない限り 3 本以上の指を使用せず、左右の手を移動させながら主に 2 本指で操作すると仮定する。この仮定のもと、次に示す規則を用いる。

- 同時刻の入力指示符が 1 個の場合
 - 直前の同じレーンに入力指示符がある場合、直前と同じ手
 - 直前の同じレーンに入力指示符がない場合で、直前に両手を使っていた場合、最も近い手
 - 直前の同じレーンに入力指示符がない場合で、直前に両手を使っていなかった場合、異なる手
- 同時刻の入力指示符が 2 個の場合
 - ロング入力指示符を保持している場合、保持中のレーンはその手、他のレーンは空いている手
 - ロング入力指示符を保持していない場合、左側は左手、右側は右手
- 同時刻の入力指示符が 3 個の場合
 - 片手のみロング入力指示符を保持している状態でない場合、左 2 レーンは左手、右 2 レーンは右手
 - 片手のみロング入力指示符を保持している場合で、保持中のレーンの左右両方に入力指示符がある場合、保持中のレーンはその手を、他のレーンは左側なら左手、右側なら右手
 - 片手のみロング入力指示符を保持している場合で、保持中のレーンの左右両方に入力指示符がある状態でない場合、保持中のレーンはその手を、他のレーンは空いている手
- 同時刻の入力指示符が 4 個の場合、左 2 レーンは左手、右 2 レーンは右手

入力指示符パターンの分類は、ロング入力指示符を含まない配置とロング入力指示符を含む配置に分けて行う。これは、ロング入力指示符を含む場合、入力の際にプレイヤーの手の拘束が発生するため、プレイヤーの手の動きがロング入力指示符を含まない場合と異なるからである。

ロング入力指示符を含まない配置については、まず左右対称な配置や左右に平行移動した配置を同一視し、配置の左右位置に依存しない分類にする。その上で、プレイヤーの手の動きを表す次の特徴を用いて配置を分類する。

- 同じ手: 同じ手が連続して使用されるかどうか
 - 交差: 片方の手がもう一方の手を跨ぐかどうか
 - レーン移動量: 各時刻間におけるレーン移動量
 - 指の本数変化: 使用する指の数が変化するかどうか
- これらの特徴により、配置に内在する手の動きを定量的に表現し、プレイヤーにとって手の動きが似ている配置同士が同一の入力指示符パターンに属するよう分類を行う。

一方、ロング入力指示符を含む配置については、ロング入力指示符による拘束の有無や手の移動量に着目し、次の特徴を用いて分類する。

- 両手ホールド: 同時刻に複数のロング入力指示符が存在するかどうか
- 交差: 片方の手がもう一方の手を跨ぐかどうか
- 最大移動量: ロング入力指示符を含まない場合と同様の基準で評価した手の移動量

この分類手法により、本研究では、譜面中の一定の時間間隔ごとの配置を、プレイヤーの手の動きに基づく入力指示符パターンとして整理・定義する。この入力指示符パターンを 4.3 節および 4.4 節の学習処理および譜面生成処理において利用することで、生成される譜面に対して、意図した操作傾向を意図的に反映させることを目指す。

4.3 学習処理

本研究では、入力指示符の配置を「タイミング」「レーン位置」「種類」の 3 要素に分解し、それぞれを個別のモデルで予測する。これにより、各要素に適した特徴表現を学習しやすくし、複雑な譜面構造を段階的に生成することを可能とする。

配置タイミングの予測には Chris, D.ら[1]による手法（以下、「DDC モデル」と呼ぶ）を用い、レーン位置および種類については、本研究で設計した機械学習モデルを用いる。これらのモデルでは、過去の入力指示符情報および時間間隔情報に加えて、配置したい入力指示符パターンに関する情報を入力として与えることで、生成される譜面に意図した操作傾向を反映させる。

レーン位置を予測するモデルでは、入力指示符が存在する時刻において、どのレーンに入力指示符が配置されるかの確率を推定する。一方、入力指示符の種類を予測するモデルでは、複数時刻にまたがる入力指示符を状態系列として表現し、各レーンにおける状態の確率を推定する。ここで、入力指示符の状態には、Chris, D.ら[1]が提案した 4 状態 on/off/hold/release を用いる。これにより、ロング入力指示符のように複数時刻にわたる入力も状態系列として扱うことが可能となるため、本研究においても採用した。

4.4 譜面生成処理

本研究では、入力された楽曲および配置したい入力指示符パターンの混合割合に基づき、譜面を段階的に生成する。譜面生成では、入力指示符の配置タイミング、レーン位置、および種類を順に決定する。

まず、配置タイミングの予測結果から時間間隔情報を算出し、これを条件としてレーン位置の予測を行う。レーン位置の予測では、過去の生成結果を入力として利用することで、過去の入力指示符配置を考慮した配置が得られるようにする。

次に、入力指示符の種類については、レーン位置の予測結果を条件として推定を行う。これにより、入力指示符の位置はレーン位置の予測結果が優先されるようにする。

また、本研究では、モデルの出力結果をそのまま採用するのではなく、入力された入力指示符パターンに属する配置と整合するよう、生成結果を調整する処理を導入する。これにより、生成された譜面が指定した入力指示符パターンの特徴をより強く反映するよう制御する。

さらに、譜面生成においては「確率最大選択条件」および「確率分布利用条件」という2つの選択条件を用いる。確率最大選択条件では、予測結果が最大となる要素と、その値に近い要素を選択する。確率分布利用条件では、予測結果を確率分布として乱数を生成し、選択する。レーン位置の選択には確率最大選択条件を用い、種類の選択には Emily, H.ら[4]にならい、確率分布利用条件を用いる。

5. 実装

4章で述べた提案手法をどのように実装したのかについて述べる。

5.1 分類処理の実装

本節では、4.2節で述べた入力指示符パターン分類手法を、実際の譜面データを用いた実装について述べる。実装は、次の手順に沿って行った。

1. 対象とする配置の決定と前処理
2. プレイヤの手の動きの推定とそれに基づく特徴量の算出
3. ロング入力指示符を含む配置の分類
4. ロング入力指示符を含まない配置の分類
5. 入力指示符グループ情報の付与

5.1.1 対象とする配置と前処理

分類対象は、4時刻4レーンにおける入力指示符の全配置とした。入力指示符の種類は通常入力指示符およびロング入力指示符の2種類のみを扱い、ロング入力指示符については開始・保持・終了の状態を区別せず、同一の入力指示符として扱った。

ロング入力指示符を含まない配置については、左右対称な配置および左右方向に平行移動した配置を同一の配置として統合した。これにより、配置の左右位置に依存しない

分類を目指した。

5.1.2 プレイヤの手の動き推定と特徴量算出

各配置について、4.2節で示した規則に基づき、プレイヤーの手の動きを時刻ごとに推定した。この推定結果に基づき、次の特徴量を算出した。

- 同じ手: 同じ手が連続して使用される回数で、各時刻間の最大値は1
- 交差: 片方の手がもう一方の手を跨ぐかどうかの値
- レーン移動量: 右方向を正として、何レーン移動するかを表す値
- 指の本数変化: 指の本数が変化する場合を1、変化しない場合を0とした値

レーン移動量および指の本数変化は、各時刻間で算出した。

また、同じ手の算出に用いる直前の時刻で使用した手の情報は、左のレーンから順に手を確認し、直前の手と現在の手を比較して決定した。次の手順に従い、各時刻の最後のレーンで決定された手をその時刻で使用した手とした。

- 最初の手なら現在の手に決定
- 右手→左手あるいは左手→右手なら両手に決定
- 左手→左手なら左手に、右手→右手なら右手に決定
- 両手→右手なら右手に、両手→左手なら左手に決定

5.1.3 ロング入力指示符を含まない配置の分類

ロング入力指示符を含まない配置については、まず「同じ手」および「交差」が同一となる配置同士を同一グループとしてまとめた。次に、所属配置数が多いグループに対して、残りの特徴量であるレーン移動量および指の本数変化の時系列的な推移に基づき、さらに細分化を行った。

配置同士の類似度を算出するため、配置間の距離行列を作成した。特徴量の推移がより類似している配置同士を近い配置として扱うため、距離計算にはDTW (Dynamic Time Warping) を用いた。

得られた距離行列に対して、UMAP (Uniform Manifold Approximation and Projection) による次元削減を行い、その結果を基にHDBSCAN (Hierarchical Density-Based Spatial Clustering of Applications with Noise) による密度クラスタリングを適用することで、入力指示符パターンのグループ分けを行った。

5.1.4 ロング入力指示符を含む配置の分類

ロング入力指示符を含む配置については、次の特徴量を用いて分類を行った。

- 両手ホールド: 同時刻にロング入力指示符が2つ以上存在しているかどうかの値
- 交差: 片方の手がもう一方の手を跨ぐかどうかの値
- 最大移動量: 各時刻間のレーン移動量の絶対値を左右の手それぞれについて合計したときの最大値が閾値以上かどうかの値

最大移動量については、ロング入力指示符を含まない配置と同様の方法でレーン移動量を算出した。これらの特徴

量の組み合わせが同一となる配置同士を、同一の入力指示符パターンとしてグループ化した。

5.1.5 入力指示符グループ情報の付与

データセットは、Chris, D.ら[1]にならない“*In The Groove*”[5]を用いた。データセットの各譜面について、4時刻ずつ入力指示符を確認し、該当する入力指示符パターンのラベルを4時刻すべてに付与した。ある4時刻の付与が完了した後、1時刻だけ移動し、再び4時刻に対してラベルを付与した。この操作を繰り返し、すべての時刻に1つ以上のラベルが付与された時点で終了した。

5.2 学習処理の実装

配置タイミングの予測には DDC モデルを、レーン位置の予測には Lane-Selection モデルを、種類の予測には Kind-Selection モデルをそれぞれ利用した。使用したデータセットについて述べた後、各モデルの詳細について述べる。

5.2.1 使用したデータセット

各楽曲には複数の難易度の譜面が存在するが、高難易度であるほど入力指示符パターンが多様であると考えられることと、すべての楽曲に譜面が存在することから、Hard 譜面のみを利用した。データセットの全 Hard 譜面から 51 譜面を取り出し、45 譜面:6 譜面:5 譜面に分割し、それぞれ訓練データ、検証データ、テストデータとして利用した。訓練データはシャッフルして使用した。

また、各時刻には最大 4 種類の入力指示符パターンのラベルが付与されているが、モデルに入力する際には、あらかじめ決定した優先度に基づいて 1 種類に限定した。これは、入力指示符パターン単体の傾向をより学習しやすくするためである。また、優先度は、学習時に利用する譜面データにおいて付与された時刻数が少ないほど高くなるように設定した。

5.2.2 Lane-Selection モデル

Lane-Selection モデルの全体図を図 5 に示す。Lane-Selection モデルは、入力指示符が存在する時刻における現在 1 時刻および直前 15 時刻分の時間間隔情報、入力指示符情報、ならびに配置したい入力指示符パターンの配置分布を入力とし、現在時刻における各レーンにおける入力指示符の存在確率を出力する。ここで配置分布とは、ある入力指示符パターンに属する全配置における 4 時刻目の 15 種類の配置の出現頻度から算出した割合である。

入力された時系列情報は、全結合層および 2 層の LSTM を通じて特徴表現へ変換され、Softmax を備えた全結合層により各配置の選択確率が得られる。この選択確率に配置分布を乗算することで、指定した入力指示符パターンに対応する配置が選択されやすくなるよう制御した。最終的に、各配置の確率を用いた加重平均により、各レーンにおける入力指示符の存在確率を出力した。

損失関数には平均二乗誤差を、最適化手法には Adam を用いた。

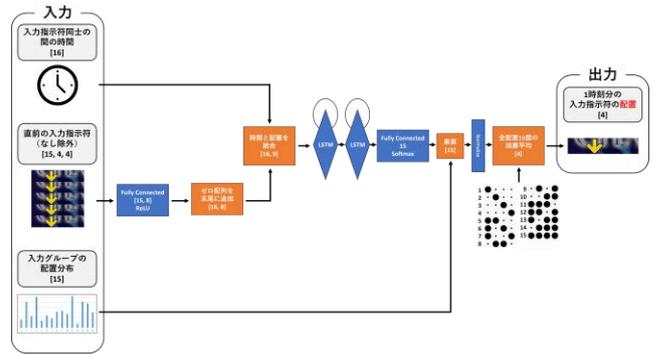


図 5 Lane-Selection モデルの全体図

5.2.3 Kind-Selection モデル

Kind-Selection モデルの全体図を図 6 に示す。Kind-Selection モデルは、入力指示符が存在する時刻における現在 1 時刻および直前 15 時刻分の時間間隔情報、入力指示符情報、配置したい入力指示符パターンの混合割合、ならびに現在時刻の入力指示符のレーン位置情報を入力とし、各レーンにおける入力指示符の状態確率を出力する。入力指示符の状態は、Chris, D.ら[1]が提案した off/on/hold/release の 4 状態とし、これらの状態系列に基づいて入力指示符の種類を決定した。

入力された時系列情報は、Lane-Selection モデルと同様に全結合層および 2 層の LSTM を通じて学習され、Softmax を備えた全結合層により各レーンの状態確率が得られる。さらに、入力されたレーン位置情報を用いて制約を与えることで、入力指示符が存在しないレーンでは必ず off が選択されるよう制御した。

本モデルは入力指示符の種類推定を目的とする分類タスクであるため、損失関数には Categorical Cross-Entropy を、最適化手法には Adam を用いた。

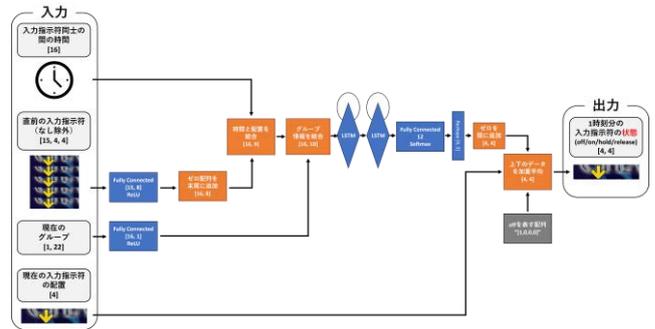


図 6 Kind-Selection モデルの全体図

5.3 譜面生成処理の実装

先に、述べた 3 つのモデルを利用し、楽曲および配置したい入力指示符パターンの混合割合から譜面を生成する際の全体図を図 7 に示す。

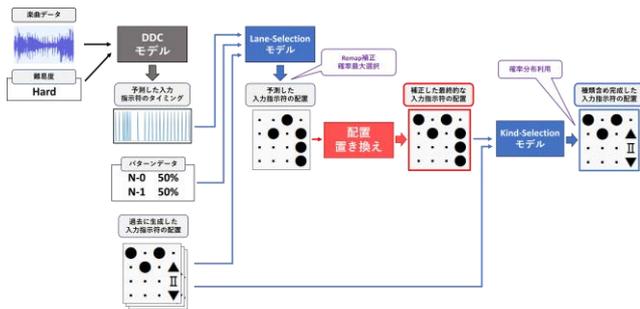


図7 譜面生成処理の全体図

DDC モデルの出力結果に基づいて入力指示符のタイミングから時間間隔情報を算出し、配置したい入力指示符パターンの混合割合とともに Lane-Selection モデルへ入力することで、レーン位置の予測結果を得た。また、入力情報の算出には直前 15 時刻分の入力指示符情報および時間間隔情報が必要となるが、初期状態では入力指示符なし、時間間隔は 0 とし、生成結果を逐次追加することで情報を更新した。

Lane-Selection の予測結果に「Remap 補正」および最大確率選択条件を適用することで、入力指示符の種類がすべて通常入力指示符であると仮定した場合の暫定的な配置を取得した。この処理を繰り返し、4 時刻分の予測が完了した時点で「配置置き換え処理」を行った。

置き換え処理後の譜面および過去の生成結果から必要な入力情報を算出し Kind-Selection モデルへ入力することで、入力指示符の種類も含めた 4 時刻分の配置の予測結果を得た。最後に、確率分布利用条件を適用し、最終的な 4 時刻分の配置の生成結果を得た。ここまでの一連の処理を繰り返すことで、1 楽曲相当の譜面を生成した。

本研究では Lane-Selection モデルと Kind-Selection モデルで別々の生成条件を用いた。Lane-Selection モデルでは、レーンごとの確率を各要素とした確率最大選択条件を用いたが、レーンによる偏りを解消するために「Remap 補正」を適用し、各レーンにおける確率の最小値が 0、最大値が 1 に近づくように補正した。Remap 補正に用いる値は、確率分布利用条件によって一度生成を行い、値が安定する開始 9.2 秒以降の生成結果から最小値および最大値を利用した。Kind-Selection モデルでは、Emily, H.ら[4]にならい、確率分布利用条件を用いた。

次に、配置置き換え処理について述べる。この処理では、入力された入力指示符パターンに属するすべての配置と暫定的な配置との類似度を算出し、最も類似度が高い配置群と暫定的な配置からランダムで選択した配置に置き換える。ロング入力指示符を含む配置については、すべて通常入力指示符に置き換えた配置を用いた。類似度は、各列および各行を順に比較し、完全に一致した回数として定義した。

入力された入力指示符パターンが複数存在する場合には、入力指示符パターンごとに最も類似度の高い配置を求

め、それらすべてを並べたうえで確率を均等に割り当てた。その後、暫定的な配置の確率が変動しないように、各入力指示符パターンの混合割合を重みとした確率を調整した。

5.4 学習条件および環境

Lane-Selection モデルおよび Kind-Selection モデルの両方において、学習率を 0.00001 に設定し、エポック数 30 で学習した。学習は CPU に Intel(R) Core(TM) i7-6700K、GPU に NVIDIA GeForce GTX 1080 を搭載した環境で実施した。

5.5 生成結果

各入力指示符パターンを混合割合 100%で入力し生成した譜面について、入力したパターンが全時刻に占める割合を算出した。ロング入力指示符を含まないパターンを入力した譜面では平均で 39.44%の割合で生成されたが、ロング入力指示符を含むパターンを入力した譜面では平均で 14.37%の割合で生成された。

また、既存譜面と同じ入力指示符パターンを入力し生成した譜面と当該既存譜面の、どちらかの譜面に入力指示符が存在する時刻における入力指示符を比較し、正解率を算出した。値は 0.07615 であった。これは、DDC モデルのみで生成した譜面でも同様に算出した値 0.06250 よりも高い値を示した。

6. 今後の展望

今後は、各入力指示符パターンの特徴を有した譜面が生成されているか、既存譜面に見られるような配置の変化が反映されているかについて、実際に生成された譜面を分析し評価することを計画している。

また、同時刻に 3 個以上の入力指示符が存在する配置において、使用した手の情報が両手ではなく右手になってしまう場合があるという問題がある。手順を見直し、これも改善を行う。

参考文献

- [1] Chris, D. et al.. Dance Dance Convolution. in International conference on machine learning. PMLR. 2017, p. 1-66.
- [2] 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. 情報処理学会論文誌. 2020, vol. 61, no. 11, p. 1718-1728.
- [3] 福永大輝, 越智景子, 大淵康成. 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. 研究報告エンタテインメントコンピューティング(EC). 2020, vol. 2020-EC-55, no. 10, p. 1-4.
- [4] Emily, H. and Matthew G.. TaikoNation: Patterning-focused Chart Generation for Rhythm Action Games. FDG '21: Proceedings of the 16th International Conference on the Foundations of Digital Games. 2021, no. 55, p. 1-10.
- [5] "Stepmania Online In The Groove". <https://stepmaniaonline.net/pack/2088>, (参照 2025-12-22).