

ピアノ宿題練習システムのための音特徴量による
生徒演奏の構成音ミスおよびタイミングミス AI 分類

Pitch Composition Miss and Timing Miss AI Classification of Student Performance
using Sound Features for Piano Homework Practice System

細田 真道¹⁾ 内山 匡²⁾ 小林 丈之¹⁾
Masamichi Hosoda Tadasu Uchiyama Takeyuki Kobayashi

笹生 恵理³⁾ 大川 耀平³⁾ 福田 成康⁴⁾ 夏川 勝行¹⁾
Eri Sasao Yohei Okawa Seikoh Fukuda Katsuyuki Natsukawa

1 はじめに

我々はピアノ教室で指導者が生徒に課す宿題の効果
を大きくするため自宅練習を補助する hiketa システム
[1, 2, 3] の実現を目指しており、そのためには指導者不
在でも演奏良否が判定できる AI 採点を実現する必要が
ある。hiketa では宿題としてある程度の長さがある曲そ
のものの練習を課すのではなく、曲の課題要素を分解
し、課題要素ごと・難易度ごとに分類した短い 4~8 小
節程度の練習フレーズ（一例を図 1 に示す）を数多く用
意して宿題として課す。そしてそれが「弾けた」か、あ
るいはどこをどのように間違ったかに応じた修正アドバ
イスを提示し、次にどの課題要素・難易度の練習フレー
ズへ進むかを決定する。これにより継続的に達成感を得
て、宿題の効果を高め、効率よく曲そのものを弾けるよ
うに補助する。

生徒の自宅環境にあるピアノが電子ピアノやキーボ
ードであれば自宅練習における生徒演奏を MIDI 収録して
詳細に比較することにより演奏ミスの分類・採点ができ
る [1, 3]。しかし、古くから普及しており数が多いと思
われるアコースティックピアノ（アップライトピアノ・
グランドピアノ）は、ごく一部に MIDI 機能を搭載した
ものもあるが多くはそのような機能を持たず MIDI での
分類・採点ができないという課題がある。

また、多くの生徒が利用できるようにするにはピアノ
の譜面台にスマートフォンまたはタブレット端末を置い
ただけの環境が望ましい。ピアノはピアノ教室の生徒な
ので所有しているはずであり、スマートフォンやタブ
レット端末は広く普及しているため所有していることが
期待できるが、その他のデバイスは生徒の所有が期待で
きない。そして、スマートフォンやタブレット端末をピ
アノの自宅練習で使用する場合、楽譜などを画面表示し
たり操作したりするためにピアノの譜面台へ置いて使用
するのが自然である。

我々は MIDI 非対応である通常のアコースティックピ
アノであっても広く AI 採点を利用できるように、生徒
演奏をマイク録音して特徴量をスペクトログラムとして
抽出し、予め登録しておいた模範演奏モデルおよび複数

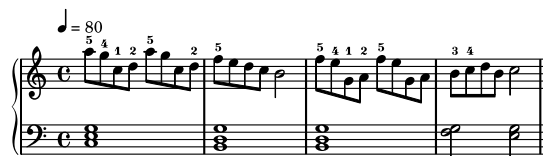


図 1 練習フレーズ「指広げ/指縮め Op. 13248」level. 12

の想定した構成音ミス（模範演奏と構成音が異なる区
間があるミス）演奏モデルのスペクトログラム同士をそ
れぞれ DTW 距離計算し、生徒演奏が登録済の模範演奏モ
デル・構成音ミスモデルのどれに一番近いかによって、
どの構成音ミスであるかを判定し分類する提案をした
[2]。しかし、この方式はタイミングミス（模範演奏と
構成音が異なる区間がなく、タイミングの伸び縮みだけ
によるミス）を検出することが困難であった。

本稿ではこの従来の構成音ミス判定に対して、新しく
打鍵間隔の比較を追加することで生徒演奏のタイミン
グミスを判定し分類できるようにする方式を提案する。こ
れによって予め登録した構成音ミス・タイミングミス双
方の既知ミスを判定し、生徒演奏がどれに近いか分類で
きるようになる。そしてその有効性を確かめるため複数
の環境における演奏を判定する評価実験を実施する。

2 関連研究

hiketa システムや課題要素の分解、練習フレーズ、
MIDI 収録による採点方式（以下、MIDI 採点）につい
ては文献 [1] に記載している。MIDI 非対応のアコース
ティックピアノに対応するためのマイク録音による採点
方式（以下、音採点）のうち、これまでに提案した生徒
演奏の構成音ミスを判定して分類する方式（以下、構成
音ミス判定）は文献 [2] に記載している。本稿ではこれ
にタイミングミスを判定して分類する方式（以下、タイ
ミングミス判定）を追加し、予め登録した構成音ミス・
タイミングミス双方の既知のミスをすべて判定し、生徒
演奏がどれに近いか分類できる方式を提案する。

音採点を実現する方法の一つとして、音を MIDI へ変
換して MIDI 採点する方法が考えられる。変換方法には
例えば FFT を用いた方法 [4] や、大規模なデータセット
による機械学習を用いた方法 [5] などがある。文献 [1]
の MIDI 採点は小節や拍を検出するビートトラッキング
が不要なのでこれらの変換結果をそのまま使用すること
も可能と思われる。しかし、採点結果は変換精度の影響
を大きく受ける上、精度を上げるため大規模なデー
タセットでの機械学習をしようとするとき、データセット
の用意が難しい、学習済みモデルが大きくなり推論だけ
でも大きな計算コストが必要になる、などの課題がある。

- 1) 東日本電信電話株式会社 デジタル革新本部 デジ
タルデザイン部。Digital Design Department, Digital
Transformation Headquarters, NTT East Corporation.
- 2) NTT アドバンステクノロジー株式会社。NTT
Advanced Technology Corporation.
- 3) 株式会社 東音企画。TO-ON Kikaku Co., Ltd.
- 4) 一般社団法人 全日本ピアノ指導者協会（ピティナ）。
The Piano Teachers' National Association of Japan.

■ミス A: 1 小節 2 拍目「レミ」と弾いてしまう



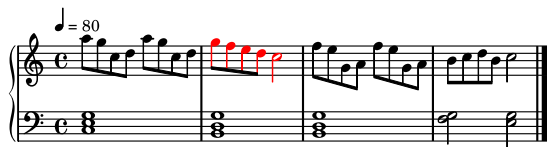
↳ 3 音目は「ド」。ゆっくり弾きましょう。

■ミス B: 1 小節 2, 4 拍目「レミ」と弾いてしまう



↳ 3 音目は「ド」。ゆっくり弾きましょう。

■ミス C: 2 小節目「ソファミレド」と弾いてしまう



↳ 5 指の音をよく確認しましょう。指を縮めますよ。

■ミス D: 1 小節 2 拍目がつかえる



↳ 3 音目は「ド」。ゆっくり弾きましょう。

■ミス E: 1 小節 2, 4 拍目がつかえる



↳ 3 音目は「ド」。ゆっくり弾きましょう。

■ミス F: 2 小節目 3 拍目がつかえる (指が足りない)



↳ 2 小節目の第 1 音は 5 指で弾いていますか。

図 2 想定される典型的な間違い演奏とアドバイス (1, 2 小節)

本稿では音を MIDI へ変換せず特徴量の抽出にとどめ、広く普及しているスマートフォンやタブレット端末などでも大きな負荷をかけずに処理できることを目指す。

ピアノなど鍵盤楽器の学習を支援する方法はいくつか提案されている。例えば鍵盤や周辺に映像を投影して支援する方法 [6] がある。本稿は一般的な生徒の自宅練習で使用できるようにするため、ピアノの他はタブレット端末やスマートフォンなど広く普及している端末のみで動作しプロジェクトなどがかりな装置構成を必要としない。もう一つの例として学習の敷居を下げるためミスの許容度を導入し、一つの曲の練習で最初は許容度を高くして成功体験を提供し段階的に許容度を下げていくことで学習を進める方法 [7] がある。本稿はある程度の長さがある一つの曲そのものを練習するのではなく曲の課題要素を分解した短い練習フレーズを数多く用意し、練習フレーズが「弾けた」体験を繰り返して継続的に達成感が得られ、その積み重ねによって曲そのものが弾けるようになる方法を採用。さらなる例としてピアノ教室で行う練習をコンピュータ上で再現しミス判定用楽譜データの作成を提案 [8] したものがあ。本稿は指導者の存在を前提とし、ピアノ教室の代替ではなく教室で課す宿題を効率よく自宅練習するシステムを目指している。また、本システムで使用する判定用モデルは統計学や情報処理などのリテラシーをほとんど必要とせずにピアノ指導者であれば容易に作成できるものを意図している。

ピアノなど鍵盤楽器の演奏では正しい指使い (運指) が重要であり大きな課題要素の一つである。そこで鍵盤を撮影するカメラを用いて演奏者の運指を認識し間違いを指摘することで学習の支援を行う方法 [9, 10] がある。本稿はタブレット端末やスマートフォンなどでの利用を想定しており、これらの多くはカメラを搭載しているが鍵盤を撮影できるように設置するのは困難であり、それ

よりは操作デバイスとしてピアノの譜面台に置いて使う方が便利であると考え。また、運指の間違いは演奏の乱れによるタイミングミスにつながるため、直接運指を認識できずともタイミングミス判定により運指の間違いを推定する方法を採用。

生徒演奏の間違いを判定するためには楽譜上のどこを演奏しているのか判断する楽譜追跡が必要となる。楽譜追跡技術はいくつかあり、例えば隠れマルコフモデル (HMM) を用いて弾き直しや弾き飛ばしに追従した楽譜追跡を行う方法 [11] がある。HMM は確率モデルであり通常のピアノ指導者が判定用モデルを作成することは困難である。本稿はある程度の長さがある一つの曲そのものの判定ではなく、短い練習フレーズの判定を行えばよいため弾き直しや弾き飛ばしが発生する余地はほとんどなく、より単純な DP マッチングを用いモデルの作成を容易なものとする。

ピアノ演奏の学習の過程を記録して間違い演奏を評価した研究 [12] がある。これも HMM を用いて間違いの検出を行っており、通常のピアノ指導者が判定用モデルを作成することは困難である。また、練習時間の間、収録し続けた連続的なデータを後から分析している。本稿は演奏後すぐに間違いを判定し、間違いに応じた修正アドバイスを提示することで次の演奏へフィードバックするとともに、どこへ進むか決定することで継続的に達成感を得て、宿題の効果を高めるシステムを目指す。

3 練習フレーズ

図 1 に練習フレーズの一つ Op. 13248 を示した。図 2 にこの練習フレーズで想定される典型的な間違い演奏と修正アドバイスを示す。これらの楽譜は MIDI 採点方式 [1] や構成音ミス判定方式 [2] で示したものと同一である。このうちミス A, B, C の 3 種類は模範演奏と構成音が異なる区間がある「構成音ミス」であり、ミス D, E,

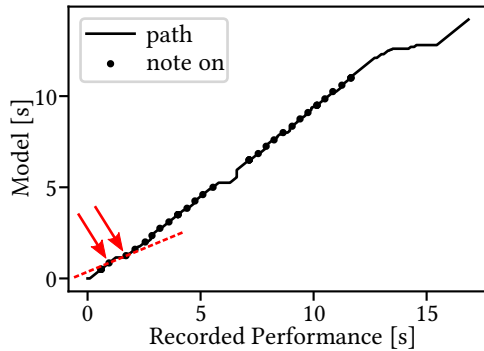


図 3 模範演奏モデルとミス D 演奏のパス (従来方式)

F の 3 種類は模範演奏と構成音が異なる区間がなく、タイミングの伸び縮みだけがある「タイミングミス」である。このうちミス F は運指を間違えたことによるタイミングミスである。

本稿の提案方式はマイク録音した生徒演奏がこれらの楽譜のうちどれに近いかを判定して分類するものであり、ミス判定となった場合は該当の修正アドバイスを示して修正を促す。

4 構成音ミス判定

構成音ミス判定 [2] の概要を示す。図 1, 2 で示した練習フレーズ Op. 13248 の正しい楽譜と構成音ミスであるミス A, B, C の楽譜から判定用のモデルを作成する。楽譜作成プログラム LilyPond [13] で譜刻 (浄書) と同時に生成した Standard MIDI File (以下, SMF) [14] をモデル SMF とする。これは MIDI 採点方式 [1] で使用するモデルと同じため共通化が可能で、楽典やピアノ指導の知識と経験があり、練習フレーズとその典型的な間違い演奏の楽譜を作成することができれば、統計学や情報処理などのリテラシーがほとんど無くても作成することができる。そして、モデル SMF 4 種類 (模範演奏, ミス A, B, C) それぞれについてソフトウェアシンセサイザ TiMidity++ [15] でモデル WAV を生成し、特徴量抽出してスペクトログラムを計算しておく。

一方、生徒演奏はスマートフォンやタブレット端末のマイクで録音したものを手弾き WAV とし、モデル WAV と同様の方法でスペクトログラムを計算する。モデル WAV から得たスペクトログラム 4 種類と手弾き WAV から得たスペクトログラムをそれぞれ時系列ベクトルとして DP マッチングの一種である DTW の計算をし、DTW 距離を求める。そして、手弾き WAV との距離が一番近いモデル WAV がミス A, B, C いずれかの場合は、該当の構成音ミスであると判定する。模範演奏モデル WAV が最も近い場合は既知の構成音ミスは存在しない、つまり「弾けた」もしくはミス D, E, F のようなタイミングミスが存在すると判定する。

5 タイミングミス判定

5.1 従来方式

構成音ミス判定 [2] では DTW の計算をしているため、片方のある時刻がもう片方のどの時刻に対応しているかの関係 (以下, パス) を得ることができ、ある程度のタイミング判定が可能である。図 3 に模範演奏モデル WAV とミス D 手弾き WAV の例としてパスを実線で、パス上に模範演奏モデル SMF におけるノート ON (打鍵) 時刻を丸いマーカーで示す。矢印はミス D で挿入

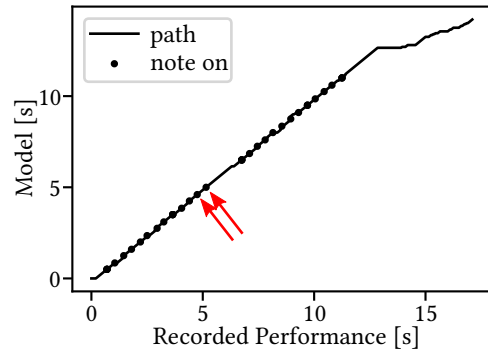


図 4 模範演奏モデルとミス F 演奏のパス (従来方式)

される余計な休符の前後に相当するマーカー、破線は矢印で示したマーカー同士の傾きを示している。破線の傾きが他より小さくなっていることから、この部分の打鍵間隔が手弾き WAV 側で大きくなっていることがわかる。これはミス D の特徴と一致するため、この部分の傾きが全体の傾きと比較して一定の閾値以下であるならばミス D であるという判定が可能と考えられる。

しかし、この方法では誤判定が多く、特にミス F 演奏では 6 割近くを誤判定した。図 4 にミス F 手弾き WAV の例を示す。矢印で示したマーカーがミス F で挿入される余計な休符の前後に相当するが、これらも含めてすべてのマーカー同士の傾きがほぼ一定で直線上に並んでおりタイミングミスが無いように見えて検出できない。この原因の一つとして、周波数解析でウィンドウをスライドさせる幅を $t_s = 0.05$ [s] としており、時間軸方向の分解能が低く本来検出すべき差が見えにくくなっていることが考えられる。また、DTW のコスト計算に双方の音量差を盛り込んでいるが、環境によって音量が異なることによってパスがズレてしまうことも考えられる。DTW は双方の変化の特徴 (極大, 極小, 変曲点, 平坦など) が同じであっても絶対値がわずかに異なるだけで対応関係が前後にズレた不自然なパスになることがあるとされる [16]。例えば、図 3 では特にマーカー間隔が大きい所でパスが直線にならずに曲っており、打鍵後の減衰や残響の違いが影響してパスがズレたのではないかと推定できる。この場合、打鍵時刻はズレていないように見えるが、環境の違いによる音量差がある場合には打鍵時刻も前後にズレる可能性があり、そのためタイミングミス検出が困難になると考えられる。

5.2 提案方式

そこで本稿では別のタイミングミス判定を追加することを提案する。構成音ミスの可能性は既に除外されており、「弾けた」かどのタイミングミスがあるか判定する。

5.2.1 音圧

まず、スペクトログラムのような時系列の多次元ベクトルではなく、時系列のスカラーとなる音圧を用いる。音圧を求める際にウィンドウをスライドさせる幅を $t_s = 0.01$ [s] とすることにより構成音ミス判定よりも時間軸方向の分解能を上げる。従来方式では判定できなかった図 4 で使用した模範演奏モデル WAV とミス F 手弾き WAV の音圧をそれぞれ図 5, 6 に示す。縦軸は音圧最大を 1.0, フロアノイズとみなせる音圧以下を 0.0 として正規化したものである。また、演奏前後の一定レベル以下の部分を無音区間として取り除いた。各打鍵タイミ

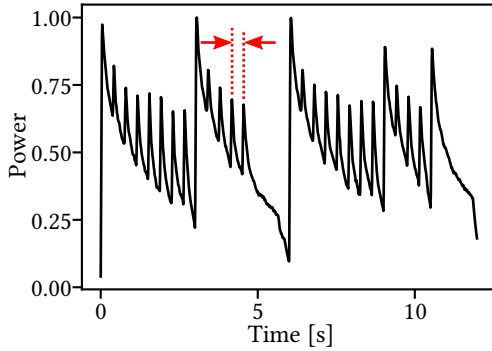


図 5 模範演奏モデルの音圧

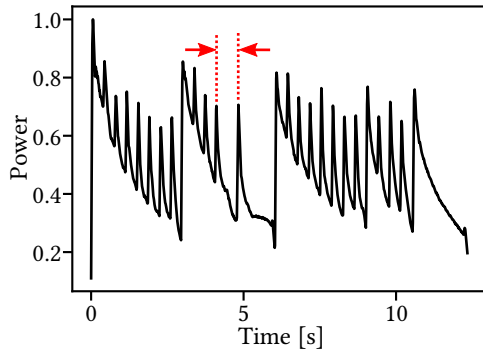


図 6 ミス F 演奏の音圧

ングがピークとして現れており、点線と矢印で示した区間がミス F で余計な休符が挿入される部分である。グラフでこの区間の長さを他の 8 分音符が並んでいる部分の打鍵間隔と比較すると、模範演奏モデル WAV ではほぼ同じなのに対してミス F 手弾き WAV では長くなっておりミス F の特徴が表れていることがわかる。

5.2.2 パス

次に、模範演奏モデル WAV と手弾き WAV の音圧同士で DDTW [16] を計算する。これは DTW のようにコストとして双方の値をそのまま利用するのではなく、微分値を利用することで変化の特徴である「形状」を考慮したパスを得ることができ、パスが不自然に前後へズレることを防ぐことができる。図 7 に提案方式で得られた模範演奏モデル WAV とミス F 手弾き WAV のパスの例を示す。矢印で示したマーカーがミス F で挿入される余計な休符の前後に相当し、破線で示した傾きが他より小さくなっていることから、この部分の打鍵間隔が手弾き WAV 側で大きくなっていることがわかる。つまり、これ検出すればミス F であることが判定できる。

5.2.3 模範演奏モデルの時刻

模範演奏モデル SMF においてノート ON 時刻が同じ音符をグループ化し、時刻順に音符グループ番号を 1 番から順番に付与したものを表 1 に示す。音符グループ番号 $i = 1, 2, 3, \dots, n$ の模範演奏モデル SMF におけるノート ON 時刻を $T_{\text{model}i}$ として、最後の音符グループ番号を n とすると、表より以下ようになる。

$$n = 26, \quad (1)$$

$$T_{\text{model}1} = 0.000, \quad (2)$$

$$T_{\text{model}n} = 10.50 \quad (3)$$

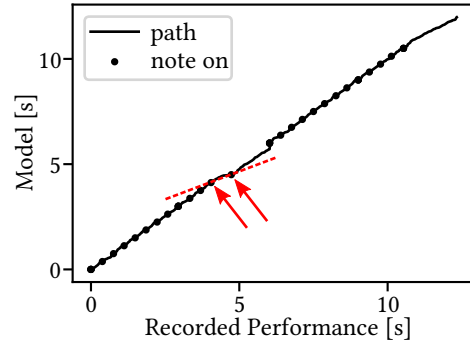


図 7 模範演奏モデルとミス F 演奏のパス (提案方式)

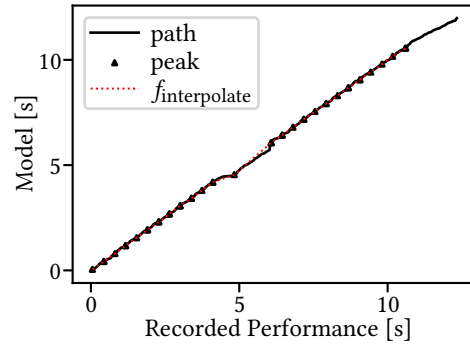


図 8 模範演奏モデルから手弾き WAV への時刻変換

5.2.4 手弾き WAV の時刻

模範演奏モデル SMF のノート ON 時刻 $T_{\text{model}i}$ から、対応する手弾き WAV の打鍵時刻 $T_{\text{foreval}i}$ を求める。この際、DDTW で求めたパスをそのまま使って時刻を変換してもよいが、双方の音圧のグラフである図 5, 6 を比較すると減衰の傾向が微妙に異なっているように見える。また、パスのグラフである図 7 では、ノート ON 時刻を示すマーカーは矢印で示した部分以外は直線的に並んでいるが、マーカー間のパスは直線ではなく曲っているところがあり、打鍵後の減衰や残響の傾向が異なることによる影響を受けているものと思われる。そこで、音圧のピークは双方で明確に表れていること、マーカーの位置そのものには特に問題が見られないことから、DDTW で求めたパス全体をそのまま使用して時刻変換するのではなく、ピークの時刻だけは DDTW のパスを用いて時刻変換し、その他の時刻については前後のピークにおける関係を線形補間して時刻変換する。図 8 にこの時刻変換を示す。実線は図 7 と同じ模範演奏モデル WAV とミス F 手弾き WAV のパスであり、三角のマーカーは WAV 上でピークを検出したところである。点線は各ピーク間を直線で結ぶ線形補間をしたものであり、これを時刻変換関数 $f_{\text{interpolate}}$ として使用する。以上により、手弾き WAV の打鍵時刻を以下の式で求める。

$$T_{\text{foreval}i} = f_{\text{interpolate}}(T_{\text{model}i}) \quad (4)$$

5.2.5 テンポ推定

模範演奏モデルと手弾き WAV は同期演奏していないため基本的にテンポは一致しない。ある程度の長さの曲であればテンポの変化を考慮する必要があるが、本稿で扱う練習フレーズは短いためテンポはほとんど変化しないとみなし、下記の式でテンポの比率を推定する。

番号	1	2	3	4	5	6	7	8	9	10	11	12	13	...	26
音名 (IPN*)	A5 G3 E3 C3	G5	C5	D5	A5	G5	C5	D5	F5 G3 D3 B2	E5	D5	C5	B4	...	C5 G3 E3
時刻 [s]	0.000	0.375	0.750	1.125	1.500	1.875	2.250	2.625	3.000	3.375	3.750	4.125	4.500	...	10.50

表 1 模範演奏モデルの音符グループ

$$R_{\text{tempo}} = \frac{T_{\text{model}n} - T_{\text{model}1}}{T_{\text{foreval}n} - T_{\text{foreval}1}} \quad (5)$$

これが 1 を超えた場合は手弾き WAV の方が速いことを、1 を下回った場合は手弾き WAV の方が遅いことを意味する。そこで一定の範囲を設定してそれを超えた場合は「速すぎます。もっとゆっくり弾きましょう。」下回った場合には「遅すぎます。もっと速く弾きましょう。」のようなアドバイスを提示することでテンポの逸脱を防ぐことができる。また、範囲内であった場合には打鍵間隔の計算で模範演奏モデルと手弾き WAV の全体的なタイミングの伸縮を補正するために使用する。

5.2.6 打鍵間隔

音符グループ番号 i と、その直前 $i-1$ の打鍵間隔の比率をテンポ補正も含めて以下の式で評価する。

$$R_{\text{IOI}i} = R_{\text{tempo}} \frac{T_{\text{foreval}i} - T_{\text{foreval}i-1}}{T_{\text{model}i} - T_{\text{model}i-1}} \quad (6)$$

これが 1 を下回る場合は手弾き WAV のこの部分の打鍵間隔が他よりも短く（速く）なっていることを、1 を超えた場合は手弾き WAV のこの部分の打鍵間隔が他よりも長く（遅く）なっていることを意味する。

5.2.7 判定

ミス D は音符グループ番号 2 と 3 の間の打鍵間隔が通常よりも大きくなるミスであると言える。同様にミス E は音符グループ番号 6 と 7 の間、ミス F は 12 と 13 の間の打鍵間隔が通常よりも大きくなるミスであると言える。そこで打鍵間隔比率に閾値を設け、1.5 を超えたらミスであると判断する。つまり

$$R_{\text{IOI}3} > 1.5, \quad (7)$$

$$R_{\text{IOI}7} > 1.5, \quad (8)$$

$$R_{\text{IOI}13} > 1.5 \quad (9)$$

を満たしたらそれぞれミス D, E, F であると判定する。

このように、タイミングミスの条件として音符グループ番号と、そこでの打鍵間隔比率の上限（つかえる、遅れるミスの判定用）・下限（詰まる、速くなるミスの判定用）の閾値を登録しておくことで、タイミングミスの判定を行う。登録した条件を満たさなかった場合は「弾けた」判定となる。音符グループ番号は楽譜上の音符を指定することで機械的に決定でき、閾値もある程度のリズム感があれば決めることができるため、楽典やピアノ指導の知識と経験のある通常のピアノ指導者であれば、こうしたタイミングミスを登録できる。

6 評価実験

提案方式の有効性を確かめるため、複数の環境で生徒演奏を模した演奏を判定する評価実験を実施する。

6.1 手弾き WAV

使用した手弾き WAV は構成音ミス判定 [2] の評価実験で使用したものとまったく同じである。環境は楽器 4 種類（グランドピアノ・アップライトピアノ・電子ピアノ-重い鍵盤・キーボード-軽い鍵盤）と空調有無 2 種類の組み合わせである。これらに対して Op. 13248 の正しい楽譜とミス A~F 楽譜の 7 種類の楽譜をそれぞれ 3 回ずつ演奏した。録音端末は 6 種類（Android スマートフォン 3・Android タブレット 1・iPhone 1・iPad 1）として 6 台同時に録音した。つまり $4 \times 2 \times 7 \times 3 \times 6 = 1008$ 個の手弾き WAV である。

これは本物の生徒演奏ではないが、熟練ピアノ指導者が作成した、生徒による典型的な間違い演奏を再現した楽譜を使用するとともに、間違いの発生原因（ミス A, B は指を広げず音高を間違えた、ミス C は指を縮めずに音高を間違えた、ミス D, E は指広げがうまくいかず遅れた、ミス F は指を縮めずに運指を間違えて遅れた）を故意になぞって間違いを発生させることにより、実際の生徒演奏を再現した演奏を収録した。なお、これら収録した演奏は熟練ピアノ指導者により生徒演奏を再現できていると判断できたものである。

6.2 結果

1008 個の手弾き WAV のうち、正しい楽譜とミス D, E, F 楽譜の 4 種類分となる 576 個の手弾き WAV について提案方式でタイミングミス判定した。すると 576 個中 572 個（約 99%）の手弾き WAV を正しく判定し、残り 4 個（約 1%）の手弾き WAV を誤判定した。

構成音ミス判定の評価実験ではタイミングミスを判定しない限定モデルでの結果は 1008 個中 998 個の手弾き WAV を正しく判定し、残り 10 個の手弾き WAV を誤判定していた [2]。この構成音ミス判定の誤判定 10 個はタイミングミスの誤判定 4 個とは別のものであった。つまり、構成音ミス判定してからタイミングミス判定をした場合の総合的な結果としては、1008 個中 994 個（約 99%）の手弾き WAV を正しく判定し、残り 14 個（約 1%）の手弾き WAV を誤判定したということになる。

6.3 考察

評価実験では多くの手弾き SMF について正解となり、提案方式の有効性を示すことができた。本評価実験では本物の生徒演奏ではなく、生徒演奏を模した演奏を収録したものであるが、間違いの発生原因そのものをなぞることにより間違い演奏を再現し、熟練ピアノ指導者により生徒演奏を再現できていると判断できたものである。つまり登録された既知のミスに近い演奏であれば本物の生徒演奏であっても問題なく判定できると考えられる。

提案方式は構成音ミス判定とタイミングミス判定を 2 段階に分けて行うため、処理時間が多く必要になってしまおうという課題がある。実用的に使用するためには処理を共通化できる部分をまとめるなどして処理時間を削減

* International Pitch Notation

する必要がある。また、スマートフォンやタブレット端末でマイク録音する際には Auto Gain Control が適用されるため音量差の検出が難しく、さらに音圧について DDTW を使用したため音量そのもののミス検出が困難になっているという課題もある。右手左手のバランスが極端に悪いなどで周波数成分が大きく異なれば構成音ミスとして検出可能ではないかと考えられるが、1 音だけ強くなった・弱くなったなどのミスは検出は難しい。そして、演奏区間を切り出す方法の検討が必要である。評価実験で使用したデータは演奏前後に無音区間としてフロアノイズだけの区間があり、その長さもすべて揃えたものである。しかし、実際に使用する場合は環境音などの影響によりフロアノイズが違ったり、録音などの操作と演奏のタイミング次第で長さも変わってしまうため、これらを考慮して演奏区間を切り出す必要がある。

未登録である未知のミス検出について、提案方式を用いた場合は登録されたタイミングミスの条件を満たさなくても、打鍵間隔比率が一定の閾値より大きい・小さい場合は未知のタイミングミスが発生しているという判定ができる。例えば、ミス D, E, F の閾値を 1.5 としたが、その他の場所で 2.0 を超えた場合や 0.5 を下回った場合などに未知のタイミングミスであると判定し、その判定をした音符グループ番号がわかるので未知タイミングミスが何小節何拍目付近で発生したかという位置も得ることができる。発生位置に応じたアドバイスや、弾けなかったときに推薦する練習フレーズなどを用意すれば、未知タイミングミスに対応することが可能となる。他にも、予め決めておいた区間に含まれる打鍵間隔比率について RMSPE や MAPE のような誤差を計算し、誤差が一定の閾値より大きい場合はその区間で演奏がもつれている、といった判定をすることも可能である。一方で、未知の構成音ミスについては DTW パス上のコストが局所的に高いなどによって検出することは可能とは思われるが、環境によってコストが大きく変わるため具体的な閾値を設定するのが困難であり、未知の構成音ミスの検出は難しいと思われる。

7 おわりに

本稿は MIDI 非対応のアコースティックピアノであっても広く AI 採点が利用できるように、生徒演奏をマイク録音してミスを判定し分類する方法について、従来の構成音ミス判定に対して新しく打鍵間隔の比較を追加することで、生徒演奏のタイミングミスを判定し分類できるようにする方式を提案した。これによって予め登録した構成音ミス・タイミングミス双方の既知ミスを判定し、生徒演奏がどれに近いか分類できるようになった。そしてその有効性を確かめるため複数の環境における演奏を判定する評価実験を実施した。その結果、打鍵間隔の比較によってタイミングミスをほぼ正しく判定することができ、構成音ミス判定と組み合わせることですべての既知ミスをほぼ正しく判定することが可能となった。

課題としては、構成音ミス判定とタイミングミス判定の 2 段階で行っているため処理時間がかかること、音量に関するミスの検出が難しいこと、未知の構成音ミスの検出が難しいこと、演奏区間を切り出す必要があることが挙げられる。今回は従来の構成音ミス判定をそのまま

利用したが、これらの課題を解消するには構成音ミス判定の改良が必要と考えられる。今後はこうした改良の他、練習フレーズや間違い演奏モデルを増やし、本物の生徒演奏も含めた手弾き WAV の収録を進めることで採点方式の改良やパラメータ調整などに取り組みたい。

参考文献

- [1] 細田真道, 最知庸, 小林丈之, 笹生恵理, 山内竣平, 野口啓之, 阪内澄宇: ピアノ宿題練習のための AI 採点方式, FIT2022 (第 21 回情報科学技術フォーラム), No. CE-007, pp. 79-84 (2022).
- [2] 細田真道, 内山匡, 最知庸, 小林丈之, 笹生恵理, 山内竣平, 野口啓之, 阪内澄宇: ピアノフレーズ練習の AI 採点のための音特徴量比較方式, FIT2022 (第 21 回情報科学技術フォーラム), No. E-015, pp. 243-246 (2022).
- [3] 細田真道, 最知庸, 小林丈之, 笹生恵理, 山内竣平, 野口啓之, 阪内澄宇: ピアノ宿題練習のための AI 採点方式ご紹介およびデモ, NTT Tech Conference 2023, Presentation Track 2 (オンライン), 入手先 (<https://ntt-developers.github.io/ntt-tech-conference/2023/>) (参照 2023-06-15).
- [4] 市来健吾: 音楽と数理 才能にたよらない耳コピ, ZENKEI AI FORUM (2020). 技術書典 9.
- [5] Hawthorne, C., Simon, I., Swavely, R., Manilow, E. and Engel, J.: Sequence-to-Sequence Piano Transcription with Transformers, *Proc. of the Int. Society for Music Information Retrieval Conf. (ISMIR)*, pp. 246-253 (2021).
- [6] 竹川佳成, 寺田努, 塚本昌彦: リズム学習を考慮したピアノ演奏学習支援システムの設計と実装, 情報処理学会論文誌, Vol. 54, No. 4, pp. 1383-1392 (2013).
- [7] 福家悠人, 竹川佳成, 柳英克: モチベーションの維持を考慮したピアノ学習支援システムの構築, 情報処理学会研究報告, Vol. 2013-MUS-98, No. 6, pp. 1-7 (2013).
- [8] 永尾謙伍, 白杵潤: MIDI 鍵盤演奏におけるテンポと強弱に着目した練習支援法に関する研究, 情報処理学会研究報告, Vol. 2014-MUS-105, No. 12, pp. 1-6 (2014).
- [9] 竹川佳成, 寺田努, 塚本昌彦: 運指認識技術を活用したピアノ演奏学習支援システムの構築, 情報処理学会論文誌, Vol. 52, No. 2, pp. 917-927 (2011).
- [10] 山川晃, 白杵潤: 画像処理を用いた MIDI 鍵盤演奏の指使い認識法に関する研究, 情報処理学会研究報告, Vol. 2012-MUS-97, No. 15, pp. 1-6 (2012).
- [11] 中村栄太, 武田晴登, 山本龍一, 齋藤康之, 酒向慎司, 嵯峨山茂樹: 任意箇所への弾き直し・弾き飛ばしを含む演奏に追従可能な楽譜追跡と自動伴奏, 情報処理学会論文誌, Vol. 54, No. 4, pp. 1338-1349 (2013).
- [12] 加藤徳啓, 中村栄太, 峯恭子, 土江田織枝, 山田昌尚: 隠れマルコフモデルを用いたピアノ練習演奏の弾き間違い分析, FIT2022 (第 21 回情報科学技術フォーラム), No. E-014, pp. 241-242 (2022).
- [13] LilyPond 開発チーム: LilyPond - みんなの楽譜作成, (オンライン), 入手先 (<https://lilypond.org>) (参照 2023-06-15).
- [14] (社) 音楽電子事業協会: 4. スタンダード MIDI ファイル 1.0, MIDI1.0 規格書 (オンライン), 入手先 (<https://amei.or.jp/midistandardcommittee/MIDIspcj.html>) (参照 2023-06-15).
- [15] 出雲正尚: TiMidity++, (online), available from (<http://timidity.sourceforge.net>) (accessed 2023-06-15).
- [16] Keogh, E. J. and Pazzani, M. J.: Derivative Dynamic Time Warping, *Proc. of the 2001 SIAM Int. Conf. on Data Mining (SDM)*, pp. 1-11 (2001).