

視線情報を活用した楽譜追跡システムの構築

寺崎 葉里^{1,a)} 竹川 佳成^{2,b)} 平田 圭二^{2,c)}

概要：本稿では DP マッチングによる楽譜追跡アルゴリズムと視線情報を組み合わせた楽譜追跡システムを提案する。従来の楽譜追跡手法の多くは打鍵情報のみで演奏箇所を推定する。そのため、同一あるいは類似した音形を多数含むような楽曲に対して、直前の演奏箇所と全く異なる箇所を演奏し始めた場合や特定箇所ですぐ打鍵と弾き直しを何度も続けて繰り返した場合は演奏箇所が確定するまでに遅れが生じたり、演奏箇所の推定精度が下がってしまう。本研究では、演奏者の思考が信頼度高く表出される視線情報に注目し、視線情報と打鍵情報を組み合わせた楽譜追跡手法を提案する。技術的な課題には、視線情報の誤差、打鍵情報の誤打鍵への対処の課題や視線情報と打鍵情報の利用優先順位があり、ヒューリスティクスの導入によって解決を試みる。楽譜追跡における視線情報の有用性を示すために、視線情報導入時と非導入時の楽譜追跡の誤り率を評価した。結果は視線情報導入時の正解率が視線非導入時の正解率の 1.4 倍に高くなった。また、視線導入時の平均推定時間が視線非導入時の平均推定時間の 0.3 倍に短くなった。従って、楽譜追跡における視線情報の有用性が証明された。

Development of a Score-Following System Using Gaze Information

TERASAKI SHIORI^{1,a)} TAKEGAWA YOSHINARI^{2,b)} HIRATA KEIJI^{2,c)}

1. はじめに

本研究の目的は、ピアノの打鍵情報と視線情報を組み合わせることで頑健かつ高速に演奏箇所を予測し、間違い、繰り返し、テンポの揺らぎ等の不確定要素に対応した楽譜追跡システムを構築することである。

複数人で演奏することで成り立つ合奏曲や重奏曲において、自分のパートの個人練習は重要である。個人練習では、他パートの音源を再生し、その演奏に合わせて自分のパートを練習することが多い。このとき、音源に合わせて演奏するだけでなく、自分の好みのテンポで演奏したり、苦手な箇所をゆっくり何度も練習したり、音楽表現のために楽曲中のある箇所のテンポに揺らぎを挿入するなど、他パートの音源のテンポを動的に変更できればより効率的に練習

できるだろう。この需要を満たすために、自動伴奏システムが開発されてきた [1], [2], [3], [4], [5], [6], [7]。自動伴奏システムは、人間の演奏に合わせて、楽譜情報に基づく伴奏をリアルタイムで適応的に演奏する。従来の自動伴奏システムは、テンポの緩急や現在演奏している箇所ですぐ打鍵をしても楽譜追跡できる。しかし、現在演奏している箇所から誤打鍵と弾き直しを続けて多数回繰り返した場合は、演奏箇所推定の誤認識率が高まる、同一あるいは類似した音形を多数含むような楽曲に対して、直前の演奏箇所と全く異なる箇所を演奏し始めた場合、ある程度演奏が進まないと演奏箇所を正確に推定することが難しい。このことから、通常の練習環境において頻繁に生じる状況に対して、従来技術よりさらに頑健かつ高速に演奏箇所を同定することが望ましい。

上述の問題を解決するために、本研究では演奏者の思考、意図、心理状態等が信頼度高く表出される視線の動き [8] に注目する。例えば、演奏者は演奏している譜面箇所を視認しているため、打鍵してない間でさえ演奏箇所が大まかに把握できる。また、視線には演奏者の意図が即座に反

¹ 公立はこだて未来大学大学院
Graduate School of Future University Hakodate

² 公立はこだて未来大学
Future University Hakodate

a) g2116028@fun.ac.jp

b) yoshi@fun.ac.jp

c) hirata@fun.ac.jp

映されるため、高速に演奏箇所を推定できる。このようなヒューリスティクスをシステムに導入することで、高精度かつ高速な楽譜追跡が実現できる。

2. 関連研究

2.1 節では本研究の要素技術となる先行研究について述べる。2.2 節では、従来の高速な楽譜追跡システムについて述べる。また、2.3 節では演奏中の視線の動きについて述べる。

2.1 Dannenberg の自動伴奏システム

Dannenberg の自動伴奏システムでは演奏と譜面との一致した音の数（ベストマッチ、Bestmatch）をコスト関数に設定し、Dynamic Programming (動的計画法, DP) マッチングによって単旋律演奏の楽譜追跡を実現している。DP マッチングとは時間伸縮を伴う音声時系列パターンのマッチングを目的として開発されたマッチング手法であり、ベストマッチを求めることで軽い跳躍や音の挿入に対応することができる。ベストマッチを算出する例として、譜面「H・A・Gis・A・C」に対して、4 音目の「A」が音抜けした「H・A・Gis・C」が演奏として入力されたときの比較を説明する (図 1)。図 1 において行列の要素がその時々々のベストマッチである。図 1 のように譜面イベントを行、演奏イベントを列とする行列を考える。演奏は時間経過に従って入力されるため、新しい演奏イベントが発生する度に新しい列を挿入していく。以下ベストマッチの計算アルゴリズムについて説明する。1 行 1 列から 2 行 1 列、...n 行 1 列、1 行 2 列、..., n 行 n 列と計算していく。この行列の 0 行目と 0 列目の値は全て 0 とする。また、n 行 m 列目の値を $D(n,m)$ と表すこととする。r 番目の譜面イベントと c 番目の演奏イベントがマッチするとき以下の中で最大のものが r 行 c 列の値となる。

- $D(r-1,c-1)+1$
- $D(r-1,c)$
- $D(r,c-1)$

r 番目の譜面イベントと c 番目の譜面イベントがマッチしないとき以下の中で最大のものが r 行 c 列の値となる。

- $D(r-1,c)$
- $D(r,c-1)$

以上のように計算から、図 1 のような行列が完成する。各列の最大値がベストマッチになるため、例では「H・A・Gis・C」が各ベストマッチ音であり、最終的なベストマッチは 4 となる。このように、DP マッチングでは音抜けがあった場合でも演奏箇所を推定できる。また、行列の最大値が複数ある場合は、一番最初の譜面イベントをベストマッチとする。

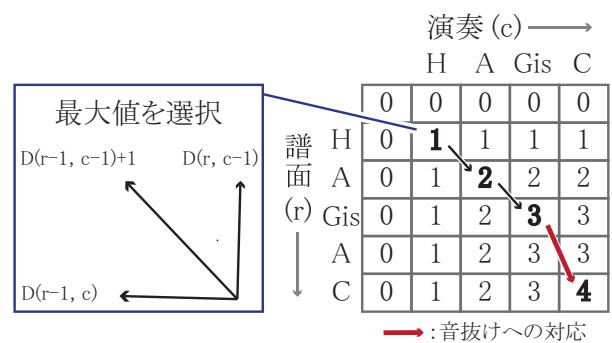


図 1 DP マッチングによる音列比較

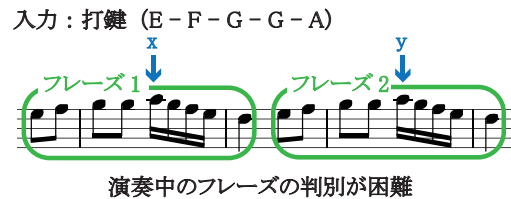


図 2 同じ音形が繰り返される例

2.2 Eurydice

楽譜追跡は確率モデルによって確率的に演奏箇所の推移を推定することで実現されることが多い。Hidden Markov Model (HMM) を用いた自動伴奏システムに中村らの Eurydice[2] がある。Eurydice は演奏生成を HMM によってモデル化し、Viterbi アルゴリズムを改良して最尤推定の計算コストを抑えて演奏箇所を探索することで、高速な演奏箇所推定を実現している。しかし、従来の自動伴奏システムは打鍵情報から演奏箇所を探索するため、譜面内に複数同じ音形が出現している場合、どのフレーズが演奏されたかを即座に特定することが難しい。例えば、図 2 において E-F-G-G-A と演奏されたとき、x と y の 2 つのいずれかの箇所にいる可能性があり、演奏箇所をどちらか一意に決めることは難しい。この研究では同じ音形を含む楽曲においても演奏箇所推定の誤り率を評価している。演奏の跳躍を含む場合と含まない場合の誤り率を比較すると、同じ音形を含まない楽曲の誤り率が 1~3 % であるのに対して、同じ音形を含む楽曲の誤り率は 6 % に増加した。同じ音形を含む楽曲は数多く存在し、実際の演奏の始まりが不確定な練習環境においてはこの問題を解決することが望ましい。

2.3 演奏と視線の関連性

安価な視線計測装置の出現により、手軽に視線情報が得られるようになり、視線の研究が盛んに行われている。また、演奏中の視線を計測する研究も存在する [9], [10], [11]。川崎らはプロとアマチュアの演奏者で楽譜の認識の違いがあるかについて調査するため、演奏中の眼球運動の分析をした [8]。この研究では読譜中の譜面上での演奏行為と注視点の位置、滞留時間の関連性に注目している。その結果、初級者が初見で譜面を見ながら演奏する場合は、譜面上の

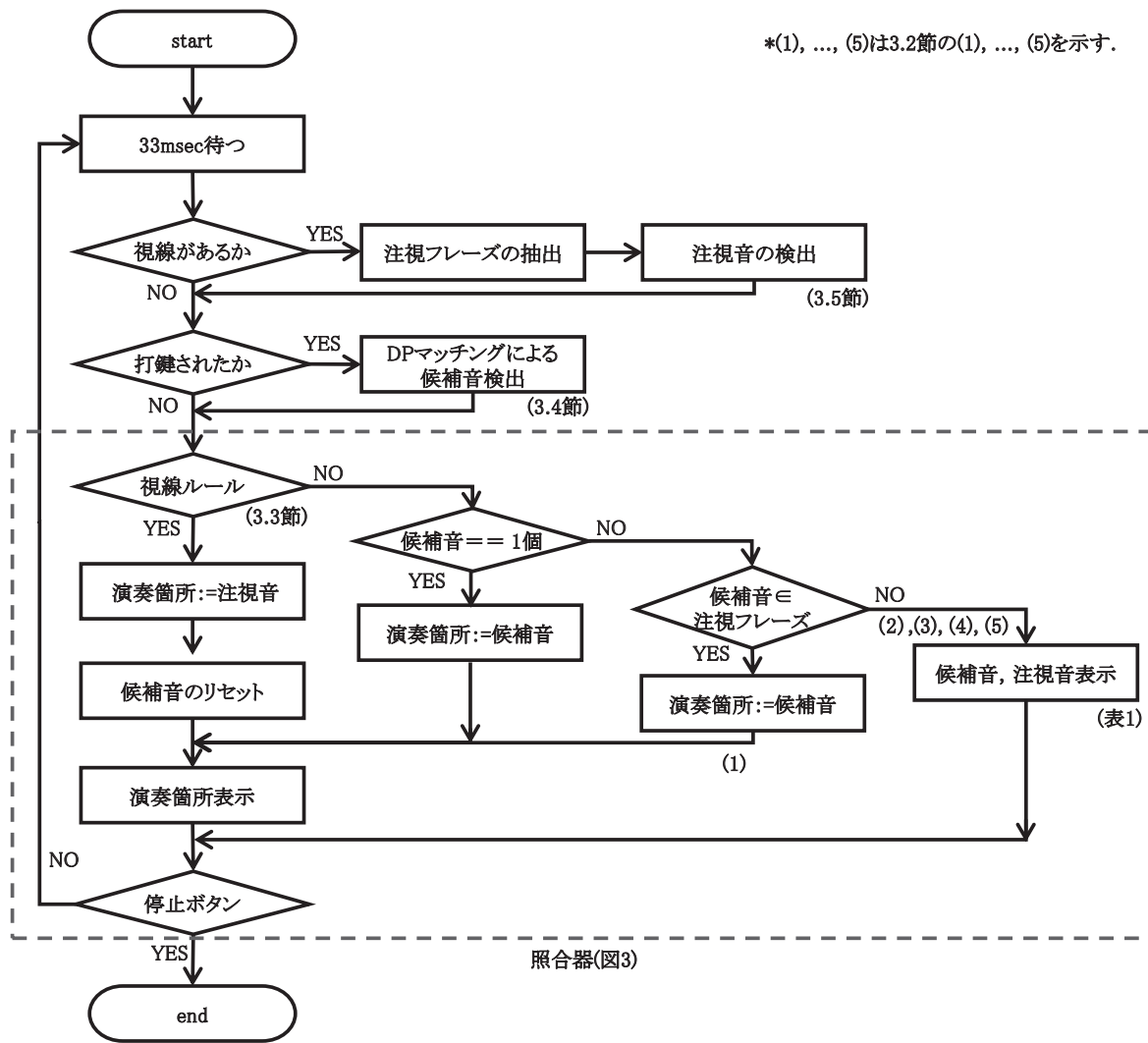


図 4 演奏箇所推定の処理手順

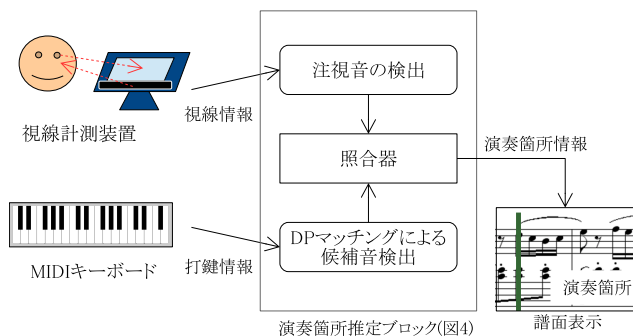


図 3 システム構成

音符を逐次的の一つずつ注視するが、練習を重ね、楽譜の構造を理解すると、初級者は音符をまとまりとして認識できるようになることがわかった。従って、ほとんどの演奏者は演奏中に音符をまとまりごとに認識し、譜面箇所を視認しているといえる。

3. 視線を活用した楽譜追跡システム

3.1 システム構成

提案システムの構成を図3に示す。本システムはMIDIデータ(打鍵位置や打鍵強度)とEyeTribe[12]による視線情報を入力とする。EyeTribeは小型かつ据え置き型の視線計測装置であるため、演奏者の演奏を妨げない。また、EyeTribeは30Hzで視線情報のサンプリングを行う。視線情報と打鍵情報の入力から演奏箇所推定を行い、演奏箇所情報として演奏箇所と候補音を出力し、譜面上に提示する。

3.2 演奏箇所推定処理

図3中の演奏箇所推定ブロックの詳細処理を図4に示す。DPマッチングにより打鍵された音列と類似した譜面の音列を全て探索し、候補音とする(3.4節)。しかし、類似度が同等の候補音が複数存在する可能性があるため、演奏箇所を一意に推定することは難しい。そこで、視線情報と組み合わせることで演奏箇所を推定する。まず、視線の

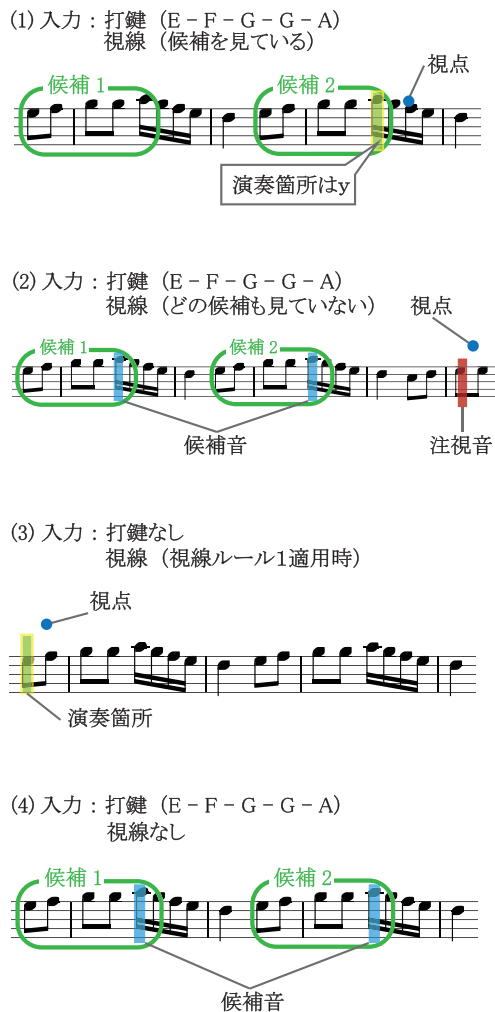


図 5 それぞれの演奏箇所推定方法が選ばれた時のシステムの動き。図中の (1), ..., (4) は 3.2 節の (1), ..., (4) を示している。(5) については特別な処理がないため、特筆しない。

位置座標から注視フレーズを抽出する。さらに注視フレーズの先頭音を注視音とする (3.5 節)。

これらの候補音、注視フレーズ、注視音の情報を照合器で組み合わせ、演奏箇所推定を行う。具体的には、視線情報、打鍵情報の有無や視線と候補音のマッチング結果に応じて演奏箇所推定方法を決定する。想定できる可能性として 5 つの場合が存在すると考え、それぞれの演奏箇所推定方法を示す。

- (1) 視線と打鍵があり、視線と一致する候補音がある場合 (図 4・図 5 中, (1)), 視線の近傍が演奏箇所である可能性が高い。なぜなら、演奏者は滑らかに演奏が進行しているときは譜面の少し先を見ている傾向をもつためである。従って、視線と最も距離の近い候補音を演奏箇所とする。これによって図 5 中, (1) のように、完全に同じ音列の候補音が複数存在しても、演奏箇所を決定できる。
- (2) 視線と打鍵があるが、視線と一致する候補音がない場合 (図 4・図 5 中, (2)), 演奏者が演奏箇所を滑らか

且つ大幅に跳躍した可能性が高い。この場合、演奏箇所を確定するのは困難である。従って、候補音が 2 箇所以上の場合には注視音と候補音を表示し、演奏箇所の確定をしない (図 5)。しかし、演奏箇所の跳躍後、演奏者が演奏箇所付近を視認しながら演奏していれば、(1) の条件により、システムは再び演奏箇所を確定することができる。ただし、候補音が一か所のみである場合は演奏箇所を一意に決定できる。

- (3) 打鍵がなく、視線のみ取得している場合 (図 4・図 5 中, (3)), 打鍵と打鍵の合間であり、直後に打鍵される可能性が高いため、演奏箇所や候補音の変動はせず、打鍵情報が入力されるまで待機する。ただし、この状態が 1 秒以上続く場合は演奏を一時中断したとして、視線情報による補正を行う。視線情報によって演奏箇所の補正をする例を図 5 中, (3) に示す。ここで視線情報による補正を行うときの条件を視線ルールと呼ぶこととする。詳しくは 3.3 節で述べる。
- (4) 打鍵はされているが、視線が取得できない場合 (図 4・図 5 中, (4)), 目を瞑って演奏していたり、譜面を見ずに演奏している。この場合、演奏箇所を確定するに充たる根拠がないとして、候補音が 2 箇所以上の場合には、演奏箇所は確定せず、注視音と候補音を表示する (図 5)。ただし、候補音が一か所のみである場合は演奏箇所を一意に決定できる。
- (5) 視線と打鍵が共がない場合 (図 4 中, (5)), 演奏者が休符の間に一瞬手元を確認したり、演奏以外のことに気を取られていることが多い。この場合は、即座に演奏箇所の補正をしようとする、休符の間に演奏箇所が変わってしまい、演奏箇所推定の精度が低下する。そのため、演奏箇所や候補音の変動はせず、視線情報か打鍵情報が入力されるまで待機する。

以上の演奏箇所推定の結果、システム上で表 1 のように音の表示をする。

3.3 視線ルール

上述のアルゴリズムは打鍵された音列と譜面の音列の類似度が高い場合には楽譜追跡できる。しかし、特定箇所連続的に誤打鍵を犯した場合など、打鍵された音列と譜面の音列の類似度が低い場合がある。その場合、正しい演奏箇所に候補音が検出されず、演奏箇所推定の精度が低下してしまう。そこで、以下の視線ルールに従って、視線による演奏箇所の補正を行う。具体的には注視音を演奏箇所とする。

視線ルール 1 : 演奏再開時への対応

演奏者が演奏を一度中断し、直前の演奏箇所とは全く違う箇所から演奏を再開するとき、直前までの打鍵情報から演奏箇所を推定することは難しい。そこで視線による演奏箇所の補正を行う。演奏者は演奏を一度中

表 1 演奏箇所と推定された音の表示

No.	視線検知	打鍵検知	候補音 \in 注視フレーズ	音の表示
(1)	○	○	○	演奏箇所を表示
(2)	○	○	×	注視音と候補音を表示
(3)	○	×	-	注視音を表示
(4)	×	○	-	候補音を表示
(5)	×	×	-	注視音, 候補音共になし

断し、演奏を再開するとき、演奏再開箇所を視認する傾向にある。このとき演奏が中断され、再開するまでの時間は1秒以上あるとする。これは演奏中に短い休止符（1小節以下程度）があり、打鍵しなかった場合、視線ルールの発動を防ぐためである。反対に、この時間を長くすると、演奏箇所推定に時間がかかってしまう。従って、1秒以上打鍵がされなかった場合、システムは演奏者が演奏再開箇所を視認していると判断し、注視音を演奏再開箇所とする。それにより、演奏再開箇所の予測や視線による演奏箇所の大幅な跳躍が可能になる。また、演奏者が全く違う譜面の箇所から演奏を再開する可能性が高いため、全ての候補音を解放し、次の打鍵を待つ。

視線ルール2：連続的な誤打鍵への対応

演奏者が現在演奏している箇所から誤打鍵と弾き直しを続けて多数回繰り返した場合は、打鍵音列と譜面音列の類似度が低くなってしまい、演奏箇所推定が困難になる。そこで、視線による演奏箇所の補正を行う。演奏者は連続的な誤打鍵があったときに、間違いを確認するため譜面の演奏箇所を視認する傾向にある。このとき、誤打鍵とは現在の演奏箇所の次の音とマッチしなかった状態を指す。1, 2度程度の誤打鍵には候補音が検出される可能性が十分にあるため、連続的な誤打鍵は3回以上とする。従って、3回以上連続でミスマッチが発生したとき、演奏者は譜面の箇所を視認するとし、注視音を演奏箇所とする。それにより、打鍵音列と譜面音列の類似度が低くなってしまいう場合にも楽譜追跡ができる。

3.4 DP マッチングによる候補音検出

打鍵された音列と譜面上の全音列を比較することで譜面上で類似した音列を検出し、それらを候補音とするアルゴリズムについて述べる。打鍵をもとにした楽譜追跡アルゴリズムとしてDP マッチング(2.1節)を採用する。DP マッチングは類似度を求めるマッチング方法であるため、誤打鍵の発生によって演奏と譜面が完全に一致しない場合でも大まかに一致していれば候補音として扱うことができる。打鍵されたとき、譜面上の打鍵された音高を全て探索し、その譜面の箇所を準候補音とする。そして、各準候補音の譜面箇所でのスコアの計算を行う。マッチングの際にス

入力：打鍵 (E-F-G-G-A) □ 準候補音 □ 候補音

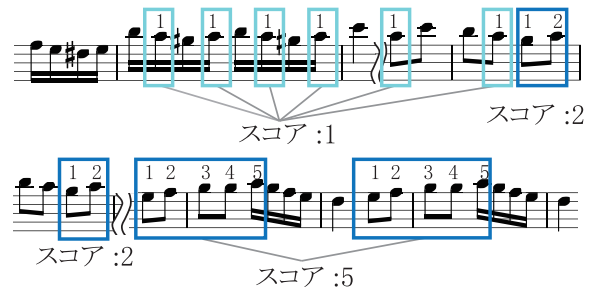


図 6 候補音と準候補音の選出

キップした音符数に応じてペナルティ (Penalty) を与えることが考えられる。例えば、譜面「H・A・Gis・A・C」に対して、入力「H・A・Gis・A」と4音目の「A」が音抜けした「H・A・Gis・C」のベストマッチは双方共に4である。しかし、前者は次に「C」が入力されれば、与えられた譜面通りの演奏になるのに対し、後者は次に打鍵される音高に関わらず、譜面通りの演奏になることはない。従って、実際には譜面通りの演奏になる可能性のある前者の方が類似度が高い。この問題に対応するために、マッチングの度にスキップした音符数をペナルティに加算し、スコア (Score) にペナルティを付与する。スコアにペナルティを付与する式を以下に示す。

$$Score = \frac{Bestmatch}{Penalty}$$

スコアが高い程ベストマッチが長く、ペナルティが小さいので、演奏箇所である可能性が高い。そこで、スコアが上位2位までの準候補音を候補音とする(図6)。図6のように、スコアが5,2,1の準候補音が存在するとき、候補音となるのはスコアが5と2の準候補音である。また、本システムに導入したヒューリスティクスとして10音以上音符をスキップした準候補音は信頼性が低いとして、準候補音から除外する。

3.5 注視音の検出

3.5.1 視線のノイズ除去

視対象から視対象に視線を移すときに発生する眼球運動として、サッケード (sacade) がある。これは視対象を認識するために、視線が常に視対象付近で高速に揺れる現象で

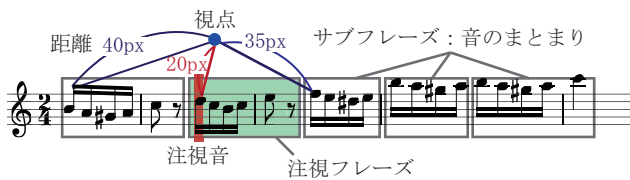


図 7 注視フレーズと注視音の検出

(a) 誤打鍵



(b) 跳躍



(c) 繰り返し



図 9 (a) 誤打鍵, (b) 跳躍, (c) 繰り返しの例

ある。サッケードによって、眼球は常に動いている。そのため、視線情報の生データをシステムに適用すると、サッケードがノイズとなり、楽譜追跡の性能が低下が心配される。そこでノイズを除去することで、安定した視線を導出し、システムに適用する。ノイズは重回帰分析によって過去1秒のデータから次点を予測することで除去する。このとき、平均値から突発的に大きく外れた値を計算に含めると予測精度が低下する。そこで、以下の値を計算対象から除外することで、予測精度の低下を防ぐ。

- 直前の値から移動距離が全体の平均値以上である場合
- 視線が認識されていない場合
- 突発的な外れ値が発生した直後に視線が同じ範囲に戻ってきた場合

3.5.2 注視音の検出

演奏者は一連の複数の音符をまとまり（サブフレーズ）として認識し、演奏している。どのサブフレーズの先頭にもならない音符から演奏を再開することは、不自然な演奏行為であることが多いため、演奏者はサブフレーズの先頭から演奏を再開することとする。この前提のもと、注視しているサブフレーズ（以下注視フレーズ）検出の処理を説明する。

(1) 計算負荷を小さくするために（譜面の段数×（1段の

小節数/2）で譜面画像を分割し、視線が存在する画面エリア（分割された譜面画像の1つの範囲とする）を調べる。

(2) 視線の存在する画面エリア内でサブフレーズ群の先頭音と視点との距離を調べ、最も視点と距離の近いサブフレーズを検出し、演奏者の注視フレーズとする。また注視フレーズの先頭音を注視音とする。図7上では各サブフレーズ群との距離（画面上なので単位はpx）が1番目のサブフレーズから40px, 20px, 35pxだったとき、最も距離の短い2番目のサブフレーズを注視フレーズとして選択している。また2番目のサブフレーズの先頭音である「D」の音を注視音としている。

4. 楽譜追跡における視線の有用性の評価

?? 本章では第3章で説明した楽譜追跡システムを用いて楽譜追跡における視線情報の有用性を評価する。4.1節で入力とする実験データ収集について述べ、4.2節で収集した実験データを用いて、視線情報導入時と非導入時で演奏箇所推定の誤り率を比較する。

4.1 入力とする実験データの収集

入力とする実験データを作成するために、譜面を読みながら演奏ができるピアノ初級者以上7名に協力を依頼した。楽曲はモーツァルトのピアノソナタ第11番第3楽章「トルコ行進曲」の冒頭16小節の単旋律を使用した(図8)。図8の枠線で囲まれている箇所のように、トルコ行進曲は冒頭16小節中に同じ音形のフレーズが3組存在し、どこのフレーズを演奏しているのか判別が難しい譜面を用いた。演奏者にはディスプレイ上に表示された楽譜を見ながら、右手のみで演奏してもらった。ここで、演奏中は自由に手元の確認をしてもよいことを伝えた。また、不確定要素を含まない演奏は楽譜追跡が容易であるため、誤打鍵、譜面内に同じ音形が含まれる箇所への跳躍、繰り返しなどの不確定要素(図9)を含んだ演奏を作為的に指示した。演奏は3回に分けて行い、それぞれ異なった内容の指示をした。それぞれの指示内容を表2に示す。1回目は2小節ごとに一音置換するように指示した。2回目は初めの4小節を演奏した後、譜面内に同じ音形をもつフレーズに跳躍することを指示した。これは跳躍する譜面の箇所を変更し、計4回行った。3回目は1,2小節目と3,4小節目と5,6,7,8小節目でそれぞれ5回ずつ繰り返して演奏することを指示した。全ての演奏は人間による演奏であるため、演奏に含めることを指示した不確定要素の他に誤打鍵や弾き直しが含まれた。また、視線認識率の低下を防ぐために、1回ずつ視線追跡装置のキャリブレーションを行った。結果として表3のような不確定要素が含まれた実験データが得られた。

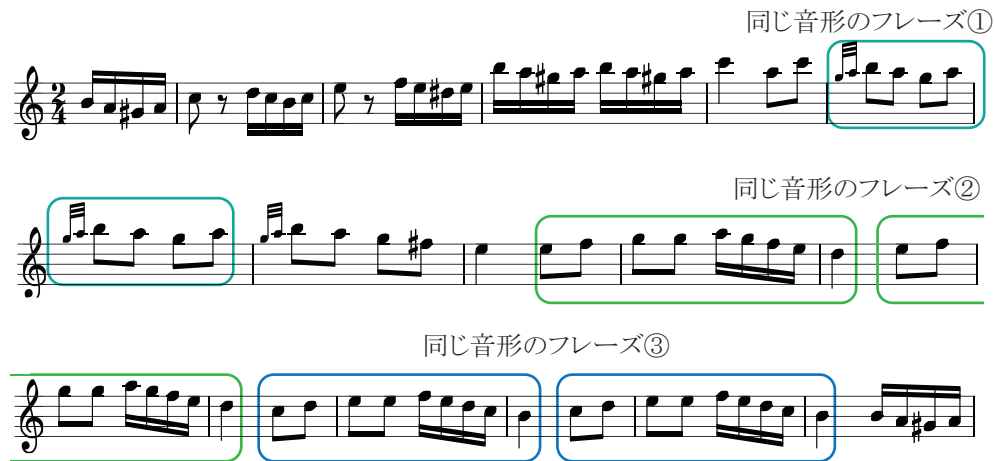


図 8 実験データ作成に使用した楽曲. トルコ行進曲の冒頭 16 小節の単旋律.

表 4 視線導入時と非導入時の楽譜追跡の正解率と平均推定時間

演奏者	視線認識率 (%)	正解率 (%)		平均推定時間 (sec)	
		視線無し	視線有り	視線無し	視線有り
A	73.0	66.6	96.6	1.30	0.69
B	72.8	67.4	98.0	1.77	0.60
C	38.1	64.3	100.0	1.65	0.50
D	70.5	73.7	100.0	1.19	0.33
E	83.4	72.7	94.6	1.30	0.40
F	68.6	72.4	100.0	1.28	0.43
G	86.3	72.41	89.7	1.82	0.66
平均	66.1	69.8	96.8	1.43	0.48

表 2 演奏へ含む不確定要素の指示

回数	指示内容
1	誤打鍵 (8 箇所)
2	跳躍 (4 回, 同じ音形の同定を要する)
3	繰り返し (3 箇所 × 5 回)

表 3 各演奏に含まれた不確定要素数 (個)

演奏者	跳躍	繰り返し	誤打鍵
A	4	15	13
B	4	15	26
C	4	15	9
D	4	15	21
E	4	15	25
F	4	15	9
G	4	15	11

4.2 結果と考察

本節では, 第 3 章で説明した楽譜追跡システムを用いて楽譜追跡における視線情報の有用性について議論する. 本システムにおける楽譜追跡に視線情報を導入した場合と導入しない場合の追従性を比較する. 入力には前節で作成した実験データを用いる. ここで, 不確定要素が発生後, 正しく演奏箇所が推定されるまでの時間を推定時間と呼ぶ. 推定時間が 2 秒以内であれば正解とし, 視線導入時と非導

入時での正解率を計算した. 正解データは人手でラベル付けすることで作成した. 表 4 に視線導入時と非導入時の楽譜追跡の正解率と平均推定時間を示す.

表 4 から, 視線認識率の平均は 66.1 % であり, 演奏者が演奏中に譜面を認識していることがわかる. 楽譜追跡の推定結果をみると, 楽譜追跡に視線情報を導入すると正解率が 1.4 倍 (69.8 % から 96.8 %) に上がるのがわかる. また, 平均推定時間が 0.3 倍に短くなるのがわかる. 結果に大きな差が認められた原因として, 演奏に譜面内に同じ音形を含む箇所への跳躍が含まれたことが挙げられる. 譜面内に同じ音形を含む箇所への跳躍が発生したときの平均推定時間が, 視線非導入時は 3.31 秒に対して, 視線を導入すると, 0.46 秒に短くなるのがわかる. 従って, 打鍵情報のみの楽譜追跡で困難であった同じ音形の同定が視線情報導入によって容易になったといえる.

次に誤打鍵の多寡による正解率と平均推定時間への影響について議論する. 誤打鍵が多い演奏者 B, E (表 3) の正解率は, 視線非導入時には 67.4 %, 72.7 %, 視線導入時には 98.0 %, 94.6 % であり, 演奏者全員の平均が 69.8 %, 96.8 % に対して大きな変化はない. 演奏者 B の平均推定時間は視線非導入時には 1.77 秒, 視線導入時には 0.60 秒

と平均推定時間が1.43秒、0.48秒であるのに対して追従が遅い。また、演奏者Eの平均推定時間は視線導入時は0.33秒、非導入時は1.19秒であり、演奏者全員の平均よりも高速に追従している。これは、演奏者Bは誤打鍵が連続して発生したため、楽譜追跡の精度が低下したと考えられる。一方、演奏者Eの演奏には誤打鍵が多く含まれているものの、連続して誤打鍵が発生することが少ないため、楽譜追跡が容易であったと考えられる。逆に演奏者Gは誤打鍵の回数が平均より少ないにも関わらず、正解率や平均推定時間の成績が良くない。これは誤打鍵を犯した直後に演奏箇所の跳躍や繰り返しをしたためである。従って、正解率と平均推定時間に影響を及ぼすのは誤打鍵の多寡ではなく、連続回数であるといえる。

5. おわりに

本稿ではDPマッチングによる楽譜追跡アルゴリズムと視線情報を組み合わせた楽譜追跡システムを提案した。技術的な課題として、視線情報の誤差や視線情報と打鍵情報の利用優先順位の課題があり、ヒューリスティクス導入によって解決を試みた。楽譜追跡においての視線情報の有用性を示すために、視線情報を導入した場合と導入しない場合で楽譜追跡の誤り率を評価したところ、正解率は1.4倍(69.8%から96.8%)に上がり、平均推定時間は0.3倍(1.43秒から0.48秒)に短くなった。

今後の課題として、本システムの実練習環境での評価がある。評価項目は以下の2点である：(1)各パラメータの妥当性、(2)実際の練習状況におけるシステムの有用性。(1)に関して、視線ルール1が発動するための待ち時間や視線ルール2が発動するための誤打鍵の回数などのパラメータは仮に決められたものであり、評価を行っていない。そのため、これらの評価を行い、パラメータが適切であるか検証する必要がある。(2)に関して、第??章では楽譜追跡における視線情報の有用性について評価したが、使用した演奏は不確定要素への評価を行うために作為的に指示されたものであり、実際の練習時の演奏とは異なる。そこで実際の練習環境におけるシステムの有用性を検証する必要がある。

また、本システムは単音に対するものであり、和音への対応がなされていない。しかし、実際の楽曲はほとんどが和音で構成されているため、和音への対応は必須である。そこで、さらに練習状況に近づけるために和音への対応を試みる。

謝辞 本研究はJSPS科研費16K12560の助成を受けたものです。

参考文献

[1] Dannenberg, R., An on-line algorithm for real-time accompaniment, In Proc. International Computer Music

- Association (ICMC), pp.193-198 (1984).
- [2] Nakamura, E., Cuvillier, P., Cont, A., Ono, N. and Sagayama, S., Autoregressive Hidden Semi-Markov Model of Symbolic Music Performance for Score Following, In Proc. International Society for Music Information Retrieval Conference (ISMIR), pp.392-398 (2015).
- [3] 中村友彦, 中村栄太, 嵯峨山茂樹, 誤り・任意の弾き直し・弾き飛ばしを含む演奏音響信号への高速な楽譜追跡, 情報処理学会研究報告音楽情報科学, Vol.40, pp.1-5 (2013).
- [4] Vecoe, B., The synthetic performer in the context of live performance, In Proc. International Computer Music Association (ICMC), pp.199-200 (1984).
- [5] Raphael, C., Music Plus One: A System for Expressive and Flexible Musical Accompaniment, In Proc. International Computer Music Association (ICMC), pp.159-162 (2001).
- [6] Cont, A., ANTESCOFO: Anticipatory Synchronization and Control of Interactive Parameters in Computer Music, In Proc. International Computer Music Association (ICMC), pp.33-40 (2008).
- [7] 前澤陽, 後藤真孝, 尾形哲也, 奥乃博, 拍長の連続性を考慮した潜在的調波配分法に基づくスコアアライメント手法, 日本音響学会講演論文集, pp.1071-1074 (2011).
- [8] 川崎智子, ピアノ演奏時における読譜の為の眼球運動, 三重大学教育学部研究紀要教育科学, Vol. 33, pp.49-66 (1982).
- [9] Kobori, S. and Takahashi, K., Cognitive Processes During Piano and Guitar Performance: An Eye Movement Study, In Proc. International Conference on Music Perception and Cognition (ICMPC), pp.748-751 (2008).
- [10] Kawase, S., An Exploratory Study of Gazing Behavior During Live Performance, In Proc. Triennial Conference of European Society for the Cognitive Sciences of Music (ESCOM), pp.227-232 (2009).
- [11] 中平勝子, 意識—無意識行動変容プロセスモデルと学習者分類—視行動に基づく技能レベル判定—, 日本人間工学会論文誌, Vol. 50, pp.90-91(2014).
- [12] The Eye Tribe, <https://theyetribe.com/> (2016年7月5日アクセス).