

# 卒業論文

(題 目)

MissionForest: 組織内外における  
協働支援のためのタスク構造化システムの試作

指導教員 白松 俊 深教授

名古屋工業大学  
工学部 情報工学科

平成 23 年 4 月 入学

(学籍番号) 24115054  
(氏名) 後藤 誉昌

(提出日: 平成 28 年 2 月 8 日)



# 論文要旨

旋律歌唱や旋律聴取における3つの処理側面として、リズム、旋律線、調性がある [1]。旋律線 (pitch contour) は旋律概形とも呼ばれ、大まかな音高の上下動を指す。調性は、調やコード進行を指す。これら3要素のうち、音楽未経験者の即興合奏を難しくするのは調性判断である。調性判断とは調の種類を判定することを指す。調性以外の旋律線やリズムについては、初心者でも比較的容易に直感的動作として表出できる。そのため、音楽未経験者が即興合奏を行いたい場合、例えば地域振興のために幅広い層の自由な参加を志向した音楽イベントなどであっても、音楽未経験者は手拍子など自由度が低い手段での参加に限られる。本研究では、ユーザの調性判断をシステムが補うことで、音楽経験に乏しい人でも調性感を損なわずに自在に即興合奏に参加できるようなシステムの開発を目指す。具体的には、ユーザが旋律線とリズムを身体動作で入力すれば、システムが背景楽曲のコード進行に応じて音高補正を行い、調性の制約を満たし不協和にならない合奏ができる演奏インタフェースの実現を目指す。そのために、(1) 直感的な身体動作による旋律線やタイミング、音の種類などの入力手法、(2) 演奏音の出力範囲の指定手法、(3) 背景楽曲のコード進行に対する調性の制約の決定手法、の3つを課題として扱う。(1)の直感的な入力手法としては、Intel RealSense 3D Camera (以下、RealSense) を用い、手の動きを認識して旋律線を入力する。RealSense は手指のジェスチャー認識が容易であり、直感的な身体動作入力に適している。本研究では特に、認識誤りの多い tap 動作の改善を試みた。(2)の演奏音の出力範囲の指定手法は、手を伸ばし画面をスクロールさせるような操作モードを設け、手の深度によってこの操作モードを切り替える。(3)の調性制約の決定手法は、能動的音楽鑑賞サービス Songle [2] の Web API からの解析情報を元に実現した。具体的には Songle に公開されている 100 曲分について統計をとり、それをもとにコードに対する調性制約を決定した。また、本研究で実装した改善版の tap 認識、調性制約に対し、評価実験を行ったところ、tap の精度は実用的なものに向上、調性制約はコードの構成音のみと比べて意図通りの演奏がしやすく、不協和が少しあるという評価となった。



# 目次

<b>論文要旨</b>	<b>1</b>
<b>第1章 序論</b>	<b>7</b>
1.1 本研究の背景 . . . . .	8
1.2 本研究の目的 . . . . .	8
1.2.1 システム構成 . . . . .	9
1.3 本論文の構成 . . . . .	9
<b>第2章 動作・開発環境と関連研究</b>	<b>11</b>
2.1 動作・開発環境 . . . . .	12
2.1.1 Intel RealSense 3D Camera . . . . .	12
2.1.2 Processing . . . . .	12
2.2 関連研究 . . . . .	12
2.2.1 Songle . . . . .	12
2.2.2 KAGURA . . . . .	13
2.2.3 Airstic Drum . . . . .	13
2.2.4 調性理解モデル PFG Tonnetz . . . . .	14
<b>第3章 直感的なリズムと旋律線の入力手法</b>	<b>17</b>
3.1 RealSense SDK . . . . .	18
3.1.1 RealSense のジェスチャー認識 . . . . .	18
3.1.2 関節の認識 . . . . .	18
3.2 持続音と減衰音 . . . . .	18
3.2.1 旋律線の入力 . . . . .	18
3.2.2 リズムの入力 . . . . .	19
3.2.3 持続音のジェスチャー . . . . .	19
3.2.4 減衰音のジェスチャー . . . . .	19
3.2.5 実行結果 . . . . .	20
<b>第4章 演奏音の発音可能な音高範囲の指定</b>	<b>25</b>
4.1 実装手法 . . . . .	26
4.2 実行結果と考察 . . . . .	26

---

<b>第 5 章 調性制約の生成</b>	<b>29</b>
5.1 調性制約の形式 . . . . .	30
5.1.1 コードに対するメロディ頻度のカウント . . . . .	30
5.1.2 周波数列の決定 . . . . .	30
<b>第 6 章 性能評価・考察</b>	<b>35</b>
6.1 実験内容 . . . . .	36
6.1.1 tap 評価実験 . . . . .	36
6.1.2 調性制約評価実験 . . . . .	36
6.2 実験結果と考察 . . . . .	36
<b>第 7 章 まとめ</b>	<b>41</b>
7.1 本研究の要約 . . . . .	42
7.2 今後の展望 . . . . .	42
<b>謝辞</b>	<b>43</b>
<b>参考文献</b>	<b>45</b>
<b>付録 A 平成 27 年度 電気・電子・情報関係学会 東海支部連合大会</b>	<b>47</b>
<b>付録 B IPSJ2016 第 78 回情報処理全国大会(掲載予定)</b>	<b>49</b>

# 図 目 次

1.1	旋律歌唱や旋律聴取の認知的処理における3つの処理側面 . . . . .	8
1.2	システム構成図 . . . . .	10
2.1	調性理解モデル PFG Tonnetz (5-limit) [9] . . . . .	14
2.2	Euler の調性理解モデル [9] . . . . .	16
2.3	Riemann の調性理解モデル [9] . . . . .	16
3.1	11種類のジェスチャー . . . . .	21
3.2	22種類の関節 [11] . . . . .	22
3.3	直感的な旋律線の入力方法 . . . . .	22
3.4	拍と portamento のタイミング . . . . .	23
3.5	tap 動作の閾値 . . . . .	23
3.6	実行結果 1 . . . . .	24
3.7	実行結果 2 . . . . .	24
4.1	音高変更範囲と tap 認識範囲 . . . . .	27
4.2	出力範囲の変更(高音) . . . . .	28
4.3	出力範囲の変更(低音) . . . . .	28
5.1	長調のコード IM の各メロディの出現頻度 . . . . .	32
5.2	長調のコード IVM の各メロディの出現頻度 . . . . .	32
5.3	長調のコード VM の各メロディの出現頻度 . . . . .	33
5.4	短調のコード IM の各メロディの出現頻度 . . . . .	33
6.1	tap 評価用紙 . . . . .	37
6.2	調性制約評価用紙 . . . . .	38
6.3	tap 評価実験結果 . . . . .	39
6.4	調性制約評価実験結果 . . . . .	39

# 表 目 次

5.1 コード変換表 . . . . .	31
----------------------	----

# 第1章

## 序論

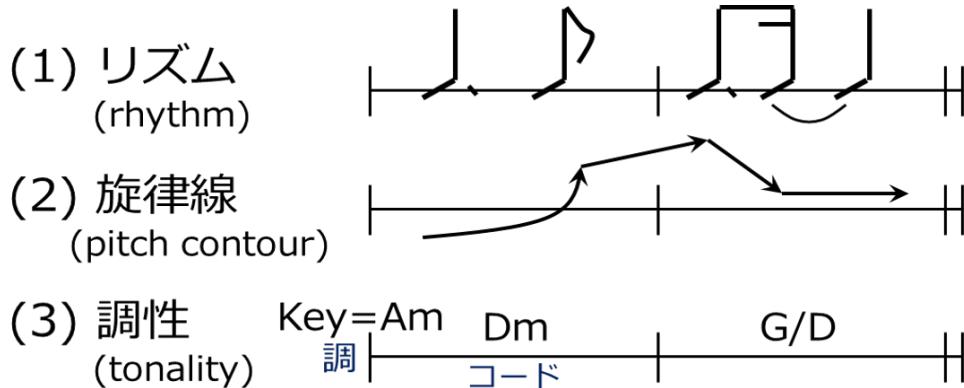


図 1.1: 旋律歌唱や旋律聴取の認知的処理における 3 つの処理側面

## 1.1 本研究の背景

音楽知識や音楽経験に乏しい人が音楽によるインタラクションを試みる際、調性感を損なわない音を発して参加することは非常に困難である。実際に音楽未経験者が自由にかつ能動的に音楽インタラクションに参加する場合として、地域社会振興のために開催される音楽イベント、アーティストのコンサート、などが考えられる。従来、音楽経験の乏しい人がそのようなパフォーマンスに参加する手段は手拍子や掛け声、リズムに合わせて体を動かす程度のことには限られている、波多野 [1] は旋律歌唱能力の発達過程や旋律聴取の認知的処理における 3 つの処理側面として、図 1.1 に示すように、(1) リズム (rhythm), (2) 旋律線 (pitch contour), (3) 調性 (tonality) があるとしている。(2) の旋律線とは音高の上がり下がりの運動パターンのことであり、初心者でもリズムやおおまかになら音が上がったか下がったかは比較的理 解しやすい。(3) の調性とは文献 [3] によれば、広義には「支配的な中心音を有する音楽系」を指し、狭義には「長短調いずれかの主和音をもつ和声的な音楽系」を指す。調性は中心音、長調/短調、音階、和音などから説明され、音楽初心者が音楽について学ぼうとする場合、調性を理解することは最も大きな障害のひとつである。よって調性は音楽未経験者が音楽活動を行おうとする際、苦手意識を持たれる原因となる。この調性に関する知識の無さが引き起こす問題点を解決することで、誰でもより充実した音楽活動をすることが容易になると考えた。

## 1.2 本研究の目的

本研究の目的は、ユーザーが直感的に操作できるように比較的知識を必要としないリズムと旋律線を身体動作で入力すれば、システムの音高補正により背景で流れている楽曲との調性の制約(以下、調性制約)に従って背景楽曲と協和を満た

### 1.1. 本研究の背景

す演奏音を出力することで、音楽知識がなくても合奏ができる演奏インターフェースの実現である。菅[4]は旋律線の手なぞりなどの身体動作が音楽理解を促進する方法として有効であるとしている。そのため旋律線の入力手段として身体動作を用いることは適切である。本研究完成時には調性感を共有した演奏参加が可能になると期待される。コンサート会場で演奏される実楽曲の調や旋法と同期させた使い方を考えると、ステージ上のアーティストと聴衆のインタラクションのあり方を変える可能性があり、さらに、学校・保育施設での情操教育や福祉施設でのリハビリテーションにも活用できる可能性があり、幼児、児童、高齢者などのQOL向上に繋がると期待される。また、旋律線とリズムを身体動作で入力することにより、ユーザーは演奏のための楽器を持ち運ぶ必要がなくなるため、演奏場所への移動距離や、スペースの制約に縛られることがない。その操作性と音高の決定手法においていくつかの問題点ある。本研究ではそれらの問題を(1)直感的な身体動作による旋律線やタイミング、音の種類などの入力手法、(2)演奏音の発音可能な音高範囲の指定、(3)調性制約の生成、の3つの課題として設定した。

### 1.2.1 システム構成

本システムのシステム構成を図1.2に示す。まず、ユーザーはモーションセンサーの前でリズムと旋律線を身体動作で入力する。本研究ではモーションセンサーはIntel RealSense 3D Camera<sup>1</sup>(以下、RealSense)を使用した、RealSenseは手指の情報とジェスチャーを検出し、認識結果を本研究で実装する合奏支援システムに渡す。システム内部ではジェスチャーによるイベント切り替えや、手の座標とあらかじめ用意しておいた調性制約から背景楽曲と不協和音にならない演奏音を出力する。

## 1.3 本論文の構成

本論文の構成を以下に示す。第2章では関連研究や関連システムと本研究の動作・開発環境について述べる。第3章では直感的な身体動作とリズムの入力手法について述べる。第4章では演奏音の発音可能な音高範囲の指定について述べる。第5章では調性制約の生成について述べる、第6章では前章までで述べたシステムの評価実験の結果をまとめ、第7章で本研究をまとめる。

<sup>1</sup><http://www.intel.co.jp/content/www/jp/ja/architecture-and-technology/realsense-overview.html>

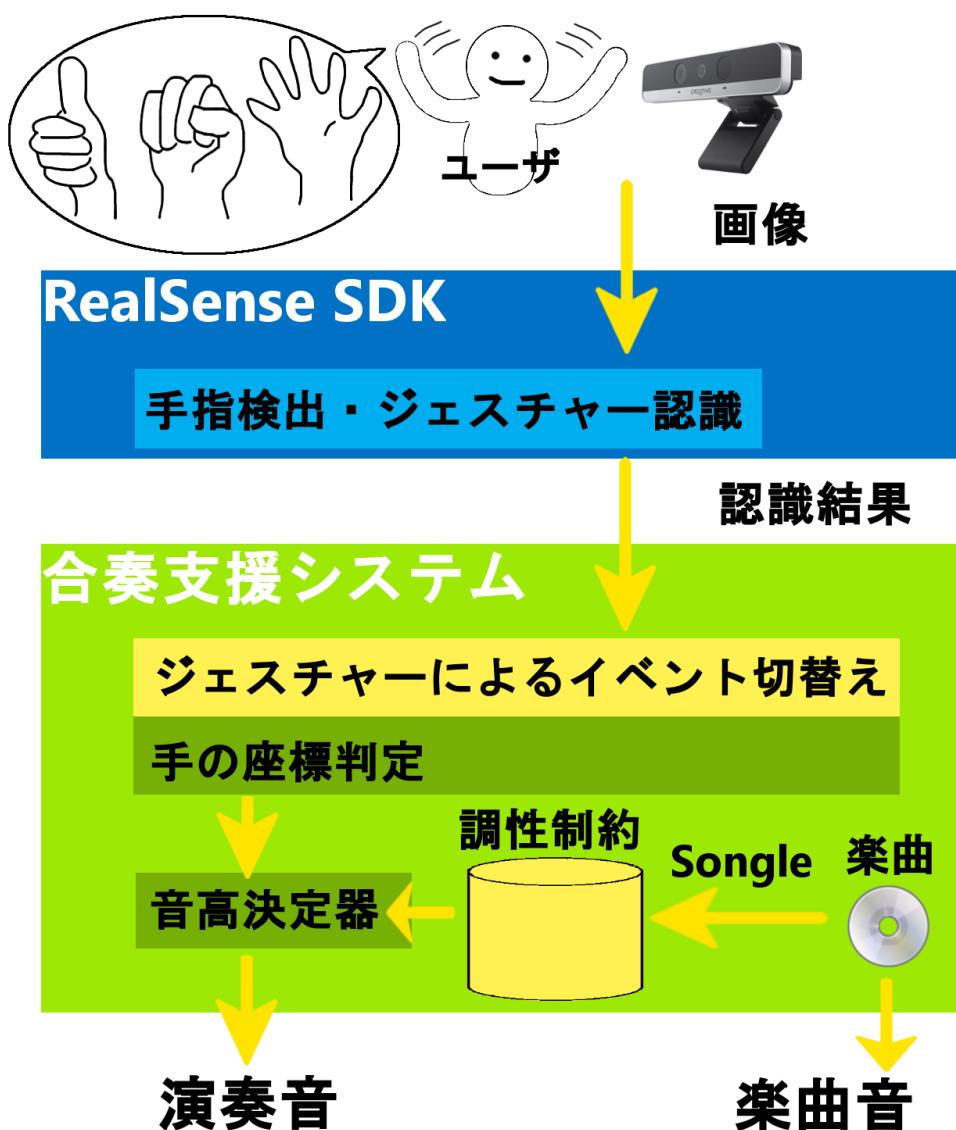


図 1.2: システム構成図

### 1.3. 本論文の構成

## 第2章

### 動作・開発環境と関連研究

## 2.1 動作・開発環境

本研究の動作・開発環境について述べる。

### 2.1.1 Intel RealSense 3D Camera

旋律線とリズムの直感的な入力手段として使用した。RealSenseは手指やジェスチャー、顔の部位や表情、心拍や音声、対象物の奥行きなどを認識することができる。さらに、今後各国の主要PCメーカーがRealSenseを内蔵したPCを発売することに賛同している。従来、Kinect<sup>1</sup>をはじめとするモーションセンサ類を使用するにはわざわざデバイスを購入しなければならず、モーションセンサ類の存在を知らなかつたり、存在を知っていても新たにデバイスを用意する手間や安価であっても購入の費用を嫌う人は一部のユーザーが使うものという認識であった。RealSense搭載PCの増加やWindows Helloなどモーションセンサでの開発経験のない一般ユーザーでも、容易に手に入ったり認知することができるRealSenseはその普及度においても期待されているデバイスである。

### 2.1.2 Processing

本システムの実装には、RealSense SDKを扱うことができ、画像や音声を比較的容易に行なうことができるプログラミング言語Processingを用いる。Processing<sup>2</sup>は、MIT(Massachusetts Institute of Technology)メディアラボに所属していたCasey ReasとBen Fryによって開発された、視覚芸術のためのプログラミング教育ように作られたJavaをベースにした開発環境である。[5]

## 2.2 関連研究

本研究と関連のある研究やシステムを紹介する。

### 2.2.1 Songle

Songle [2]とは、音楽をサビ、メロディ、コード、ビートなどの構造を表示しながら鑑賞できるサービスで、ピアプロ<sup>3</sup>やSoundCloud<sup>4</sup>の楽曲ページ、ニコニコ動

---

<sup>1</sup><http://www.xbox.com/ja-JP/kinect>

<sup>2</sup><https://www.processing.org/>

<sup>3</sup><http://piapro.jp/>

<sup>4</sup><https://soundcloud.com/>

画<sup>5</sup>や YouTube<sup>6</sup>などにアップロードされている合法的に視聴可能な楽曲を音楽理解技術により自動解析し A メロ・B メロ・サビといった繰り返し構造や、ビートの位置・コード進行・ボーカルの音高などを自動的に解析して見ることができる。さらに Songle 上では複数のユーザでコンピュータの自動解析による誤差を訂正することで、解析結果の精度を高めている。また、Songle 解析結果を JavaScript から操作し、WEB 上で利用するための Songle Widget API<sup>7</sup>を用いることによって解析結果を JSON ファイルとして参照することができる。

### 2.2.2 KAGURA

中村ら [6] は RealSense を利用した音楽演奏アプリ KAGURA [7] を開発した。KAGURA は RealSense を使用することにより RealSense の特徴であるジェスチャー認識機能を利用し、身体動作で音の高さを、ジェスチャーで細かい設定や操作を指定することができる。出力される音は不協和音にならないようになっているため、適当に体を動かしたり、音楽演奏経験のない人でも音楽を演奏することができる。また、KAGURA には録画機能と YouTube へのアップロード機能が付いているため、気に入った演奏をすぐにインターネット上に公開することも容易に行えるようになっている。

### 2.2.3 Airstic Drum

菅家ら [8] は、物理的に打面のあるドラム楽器（以下、実ドラム）の問題点を解決するために、実ドラムの補助として使用頻度の少ない打楽器に対して仮想ドラムを適用することで実ドラムと仮想ドラムの両方の利点を持った Airstic Drum を設計した。この仮想ドラムの実装は無線通信機能を持つ加速度センサを搭載したドラムスティックから得られる加速度、角速度に閾値を設定し音の出力タイミングとしている。具体的には、加速度が設定された閾値（以下、基準強度）を超えるか基準強度を下回るまでの経過時間の実ドラムと仮想ドラムの違いを識別して仮想ドラムの叩打タイミングとしている。また、仮想ドラムの叩打と演奏音の出力タイミングの時間差を測定するため、60bpm, 90bpm, 120bpm のテンポでメトロノームに合わせて仮想ドラムの叩打を繰り返し、メトロノームのクリック音と MIDI 音源にメッセージが送信されるタイミングの時間差を計測した。

---

<sup>5</sup><http://www.nicovideo.jp/>

<sup>6</sup><https://www.youtube.com/>

<sup>7</sup><https://widget.songle.jp/>

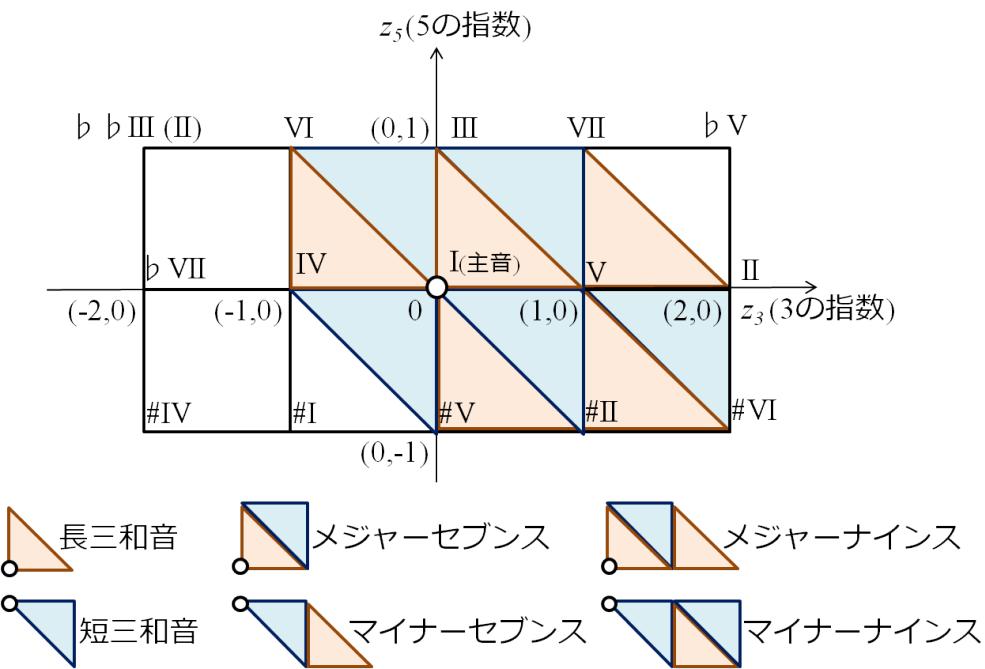


図 2.1: 調性理解モデル PFG Tonnetz (5-limit) [9]

## 2.2.4 調性理解モデル PFG Tonnetz

白松ら [9] は調性理解モデル PFG Tonnetz (Prime Factor-based Generalized Tonnetz) を考案した。ある調の主音の周波数と協和音程の周波数比は、

$$f = \left( \prod_{p \text{ は } n \text{ 以下の素数}} p^{(z_p)} \right) \cdot f_{\text{tonic}} \quad (z_p \text{ は整数}) \quad (2.1)$$

のような素数の積で表すことができる。白松らは  $n = 5$  と置いた上で、オクターブ隔たった音は音楽的に等価であることを考慮し、2 の指数  $z_2$  軸をなくして  $z_3 z_5$  平面へ投影すると、

図 2.1 のように  $z_3 z_5$  平面に配置される。この平面上の根音  $(a, b)$  に対して以下の ような整数格子点列  $\text{chord}(a, b, \delta, m)$  を考える。

$$\text{chord}(a, b, \delta, m) = \left[ (a, b) + \sum_{i=0}^k \delta(i) \right]_{k=0,1,\dots,m} \quad (2.2)$$

$$\delta_{\text{maj}}(i) = \begin{cases} (0, 1) & (i \text{ が奇数のとき}) \\ (1, -1) & (i \text{ が偶数のとき}) \end{cases} \quad (2.3)$$

$$\delta_{\text{min}}(i) = \begin{cases} (1, -1) & (i \text{ が奇数のとき}) \\ (0, 1) & (i \text{ が偶数のとき}) \end{cases} \quad (2.4)$$

根音  $(a, b)$  を調の主音  $(0, 0)$  で置き換えて考えると、明るい長音階の構成音 (I, II, III, IV, V, VI, VII) が原点の上側  $0 \leq z_5 \leq 1$  に分布し、暗い短音階の構成音 (I, II, #II, IV, V, #V, #VI) が原点の下側  $-1 \leq z_5 \leq 0$  に分布するのが見て取れる。

このモデルは、音名、階名、コード名のようなシンボルを前提としては用いておらず、周波数比に基づく認知的原理のみから導かれる。まとめると、導出の過程で現れた以下の表現形式は、上記の認知的原理および導出されたモデルの観察から自然に導かれたものである。

- 整数格子点  $(z_3, z_5)$ : 音階の構成音、階名
- 三角形  $[(a, b), (a, b+1), (a+1, b)]$ :  
整数格子点  $(a, b)$  を根音とする長三和音
- 三角形  $[(a, b), (a+1, b-1), (a+1, b)]$ :  
 $(a, b)$  を根音とする短三和音
- 整数格子点列  $\text{chord}(a, b, \delta_{\text{maj}}, m)$ :  
 $(a, b)$  を根音とする長和音
- 整数格子点列  $\text{chord}(a, b, \delta_{\text{min}}, m)$ :  
 $(a, b)$  を根音とする短和音

この PFG Tonnetz は数学者 Leonhard Euler が 1739 年に考案した調性理解モデルをもとに、音楽学者 Hugo Riemann が 1880 年に発展させた Tonnetz というモデルに似ている Tonnetz は 1980 年代以降、数学的に定式化された新リーマン理論 (NeoRiemannian theory) へと発展し、様々な拡張が行われた [10] が基本的には図 2.2, 図 2.3 のように音名同士を繋いだモデルであり、調の主音は表現できていない。

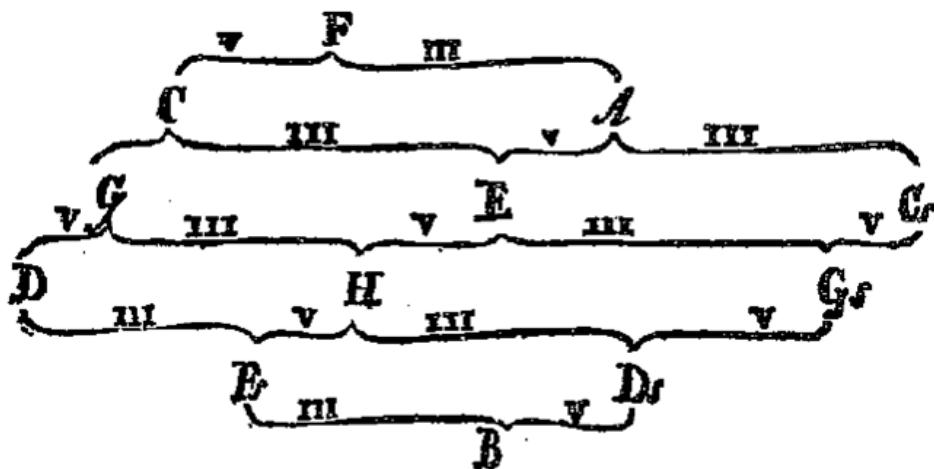


図 2.2: Euler の調性理解モデル [9]

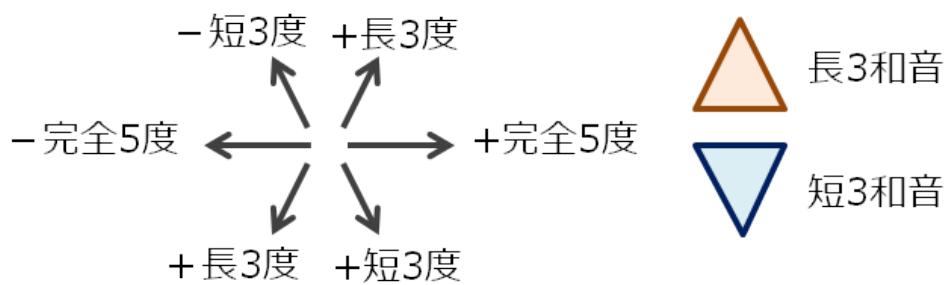
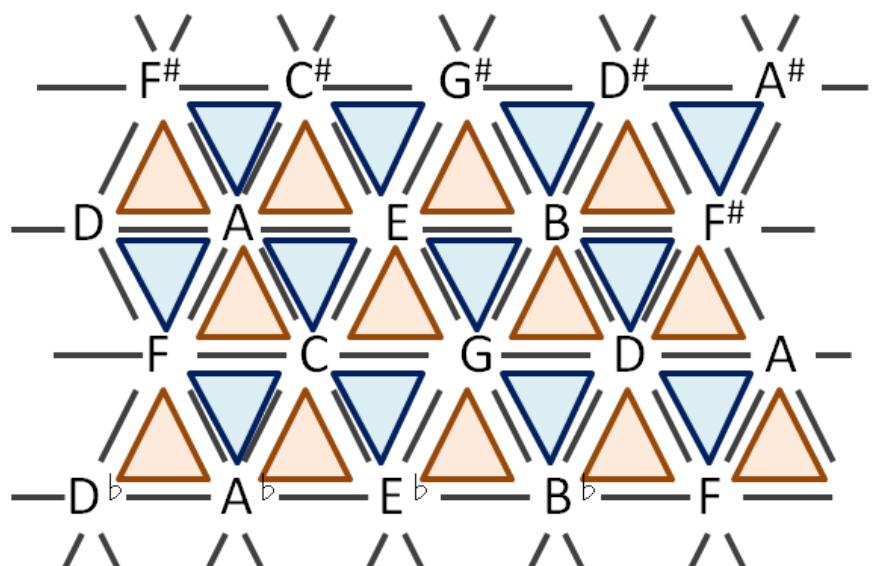


図 2.3: Riemann の調性理解モデル [9]

## 2.2. 関連研究

## 第3章

### 直感的なリズムと旋律線の入力手法

本章では身体動作で直感的にリズムと旋律線を入力する手法について述べる。

## 3.1 RealSense SDK

RealSense を扱うための RealSense SDK が提供する機能のうち、本研究で使用したものについて述べる。

### 3.1.1 RealSense のジェスチャー認識

RealSense SDK では、図 3.1 に示す 11 種類のジェスチャーが定義されている。しかし、あらかじめ定義されているジェスチャーの多くは、認識精度に問題があり、認識漏れや遅延が発生する。音楽を演奏するインターフェースの操作手段としてジェスチャーを使用するにはこの遅延が大きな障害となる。本研究ではジェスチャー認識に加えて、手指や手のひらの座標、開閉度などのパラメータを利用して認識精度を向上させた。

### 3.1.2 関節の認識

RealSense カメラでは図 3.2 に示す 22 種類の手のひらの関節を認識することができる。本研究では図 3.2 の Middle Fingertip の座標を指先の座標、Palm の座標を手のひらの座標として扱った。

## 3.2 持続音と減衰音

持続音とはバイオリン、笛などのような音量が一定のまま鳴り続ける音で、減衰音とはピアノ、鉄琴、木琴などのような音量が減衰していく音である。本研究ではユーザの演奏表現にバリエーションを提供するために持続音と減衰音の両方を扱う。よってユーザーが好きな時に持続音と減衰音を切り替えられるように、図 3.1 のうち thumb\_up 動作をすると演奏音が持続音か減衰音に切り替わるように割り当てた。本研究では左右の手で出力される演奏音は独立しており、2 和音まで出力可能である。

### 3.2.1 旋律線の入力

図 3.3 に身体動作による旋律線の入力の概要を示す。ジェスチャーで音の種類を持続音に指定しユーザーは RealSense の前で動くと指先の座標の時間的変化がそのまま旋律線として入力される。Processing 側では画面を現在発音可能な演奏音の数

## 3.1. RealSense SDK

で等分割した領域を用意して、インターフェースの画面に色分けして表示する。手指の座標がどの領域に存在するかにより、その領域とひも付けられた背景楽曲の調性制約を満たす周波数の音が演奏音として出力される。

### 3.2.2 リズムの入力

持続音を出力する際に現在演奏している音の領域と離れた領域の音を出力する場合、手を移動させた時に通った領域の音まで出力してしまう。そこでSongle Web APIから取得した背景楽曲の拍に合わせて点滅する円を画面に表示させ、リズムを取りやすくするとともに、円が表示されてないタイミングのとき演奏音をportamentoを設定することで、離れた領域の音を出力するときに間の領域の音を出力するのを防ぐ。本研究では点滅する円に合わせるようにして手を動かすことをリズムの入力とした。図3.4に拍とportamentoのタイミングを示す。曲のビート一拍分を半分に分けた時の前半部分の間に円を表示させ、後半部分の間に円を非表示にしてportamentoを設定する。後半部分の間指先を動かして領域を移動すると、出力音は滑らかに変化する。

### 3.2.3 持続音のジェスチャー

持続音のオンセットとオフセットには、それぞれ図3.1のspreadfingersとfistを割り当てた。これらのジェスチャーの精度を向上させるため、指の開閉度というパラメータ(0~100)を利用した。ジェスチャー認識の条件に開閉度が90以上の時、spreadfingers動作を、1fあたりの変化量が-10を下回った時、fist動作をするように設定させた。

### 3.2.4 減衰音のジェスチャー

減衰音のオンセットには図3.1のtap動作を割り当てた。本研究の持続音は1秒間だけ出力されるので減衰音のオフセットにジェスチャーは割り当てない。tap動作をしたとき指先の座標が存在する領域に対応した周波数の音が出力される。tap認識の精度を向上させるため、指先と手のひらの速さと移動距離に閾値を設定したときtap動作をしたと認識させた。図3.5に閾値を設定した場所を示す。閾値は経験的に指先のz座標のカメラ方向への移動距離が20mm以上移動した時、指先の3次元移動速度が1.8m/s以上、手のひらの移動速度が0.6m/s以上1.5m/s以下の時tap動作とした。本研究のtap動作はRealSense SDKのtap認識の代わりに使用しているため、どちらが優れているかの評価実験を行い、第6章でまとめた。

### 3.2.5 実行結果

図3.6, 図3.7に実行結果を示す。画面下の拍によって点滅する円である。指先の小さい円が指先の座標を表し、領域の色が濃くなっているところが現在指定している高さの領域である。赤になるほど音が高く、緑になるほど音が低くなる。

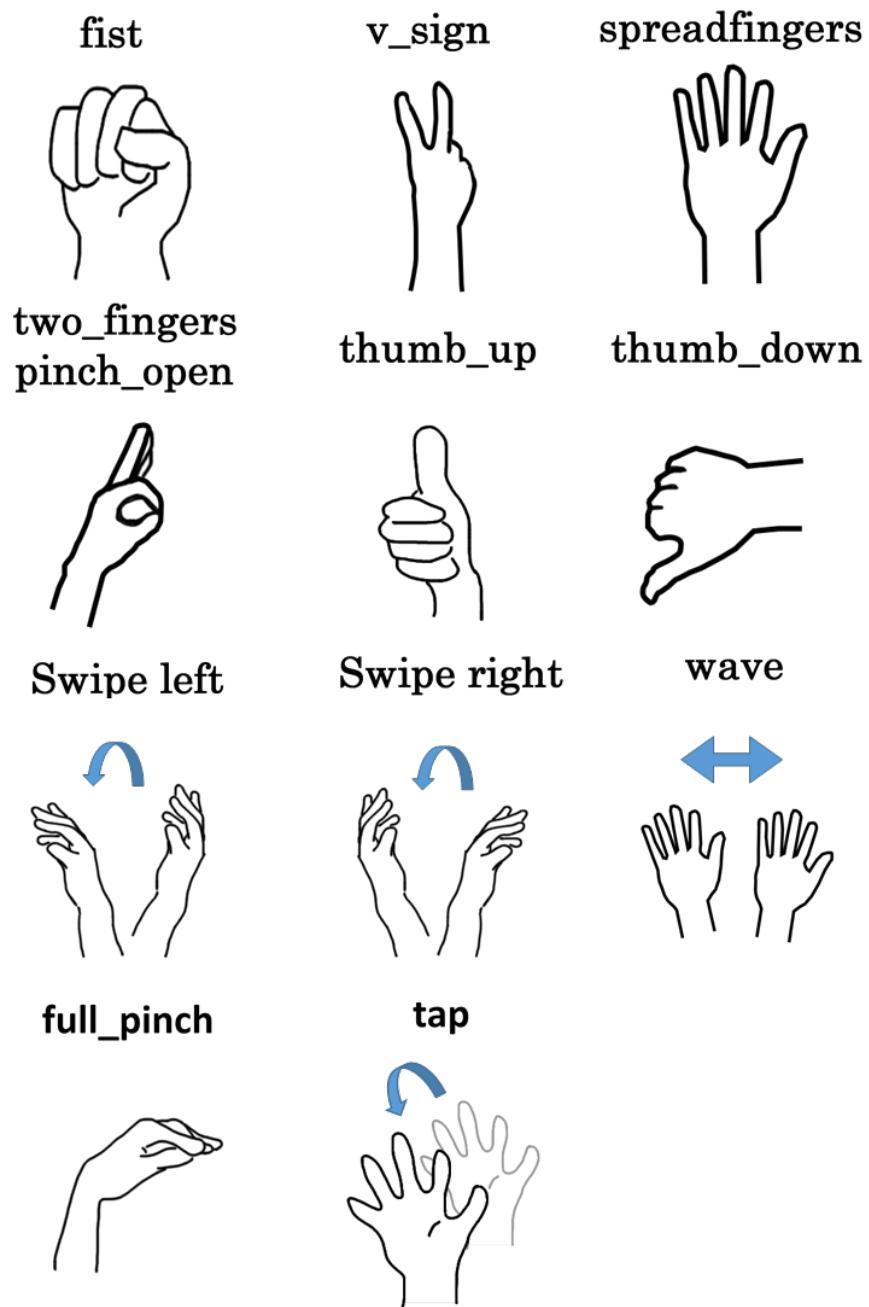


図 3.1: 11 種類のジェスチャー

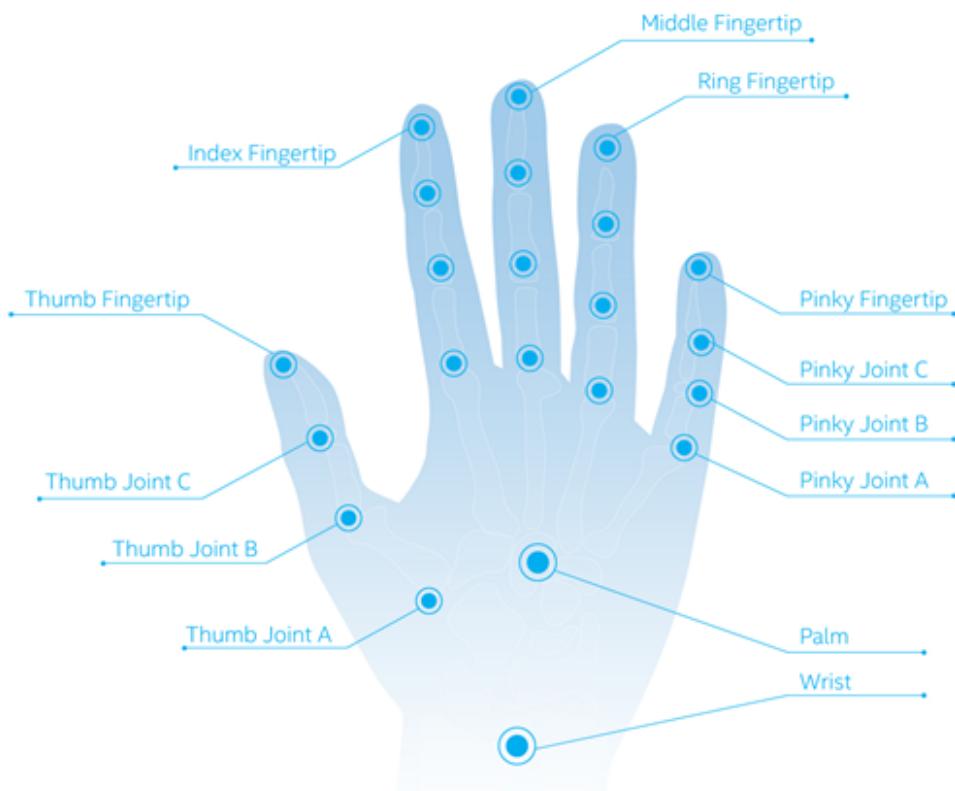


図 3.2: 22 種類の関節 [11]

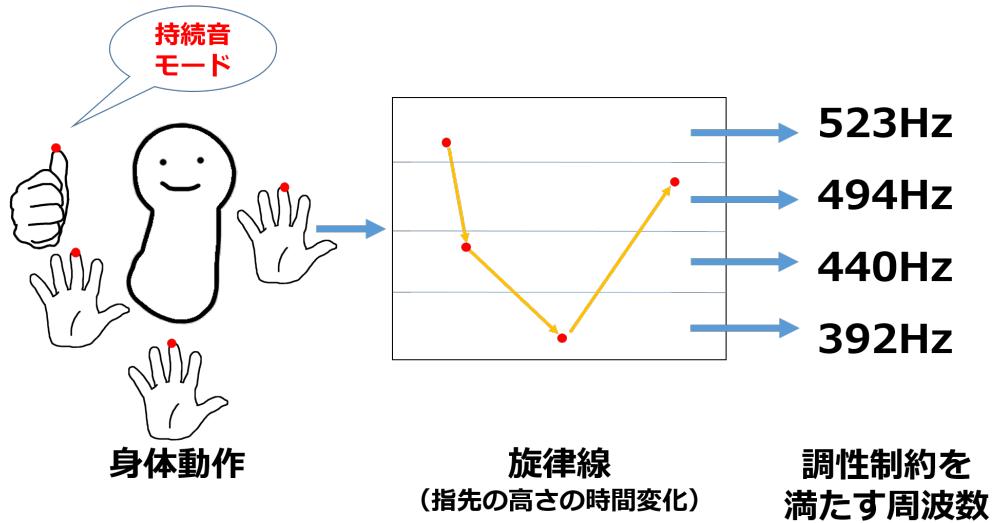


図 3.3: 直感的な旋律線の入力方法

### 3.2. 持続音と減衰音

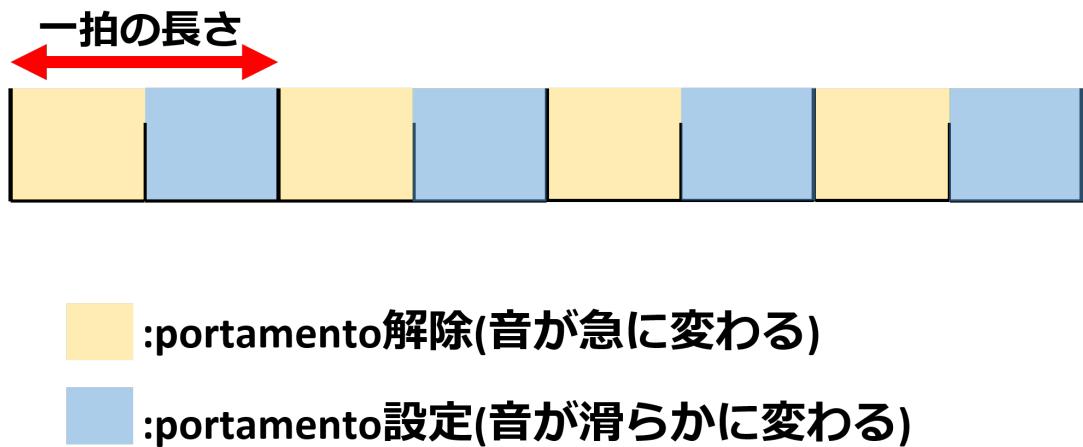


図 3.4: 拍と portamento のタイミング

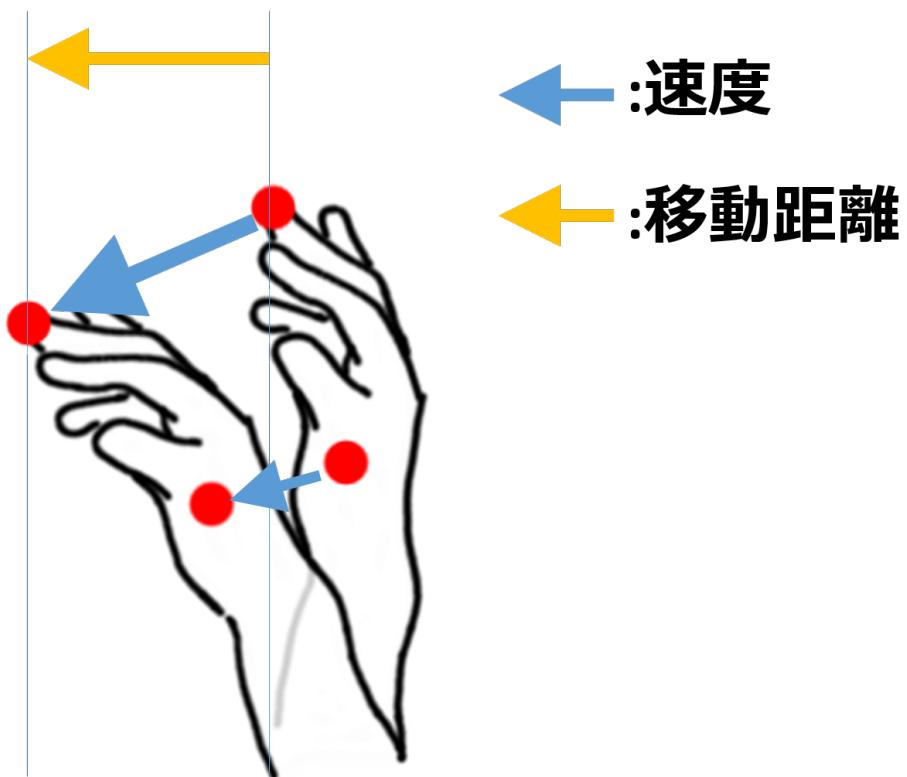


図 3.5: tap 動作の閾値

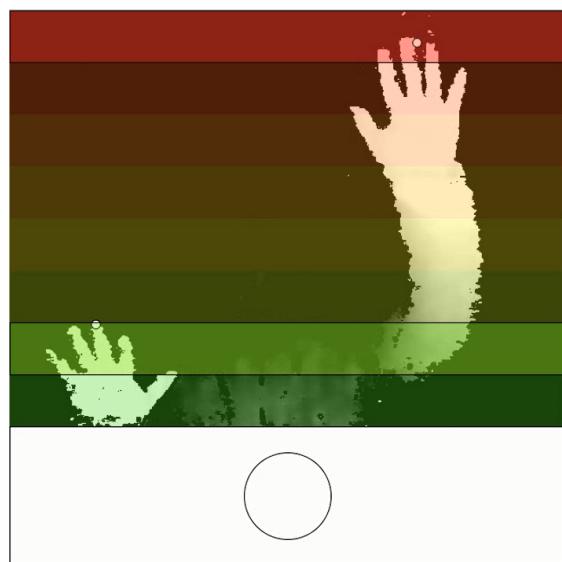


図 3.6: 実行結果 1

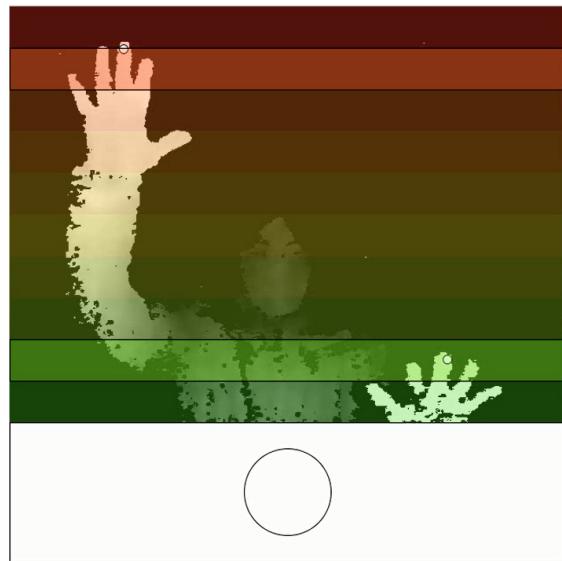


図 3.7: 実行結果 2

### 3.2. 持続音と減衰音

## 第4章

# 演奏音の発音可能な音高範囲の指定

演奏音の発音可能な音高範囲(以下、音高範囲)を指定する手法について述べる。

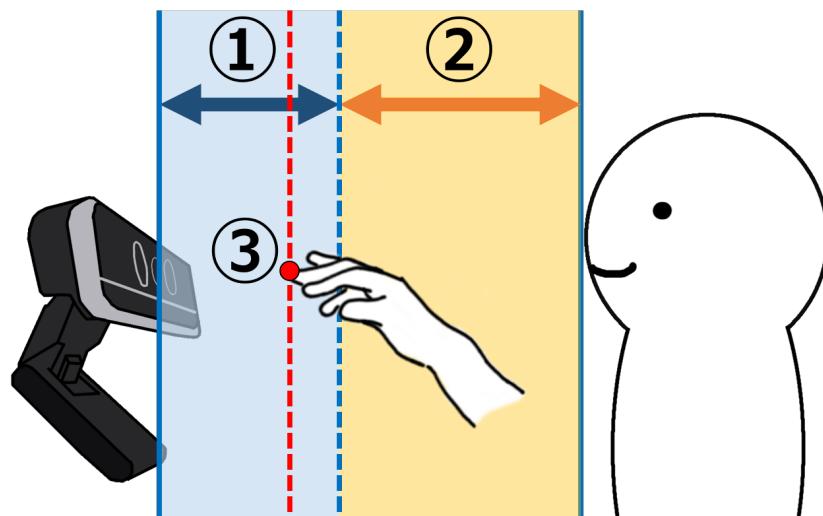
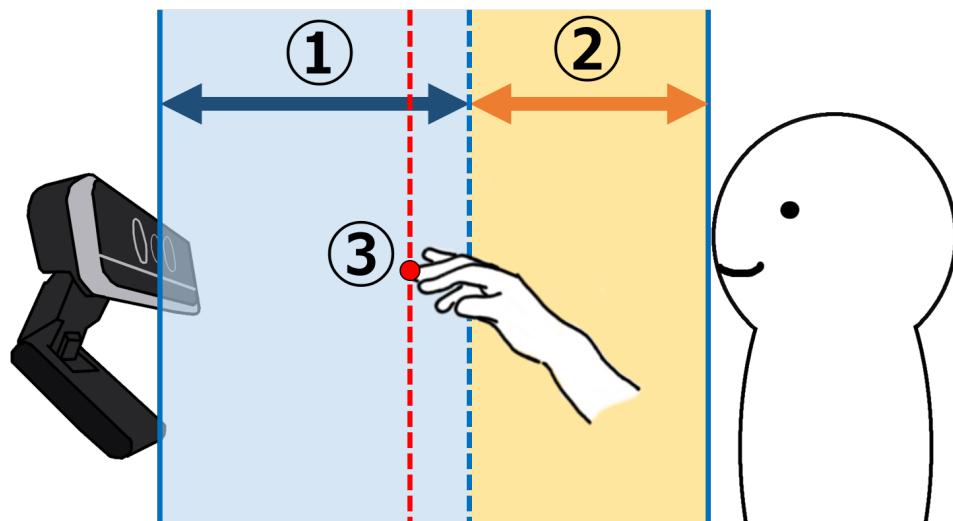
## 4.1 実装手法

RealSenseで扱えるカラー画像の最大サイズは $1920\times 1080$ である。しかし本研究では主に深度画像を扱うため深度画像の最大サイズ $640\times 480$ がインターフェースの画面サイズとなる。本研究のインターフェースは現在発音可能な音を、画面のサイズを音の数で均等に分けて色分けされた領域で表示している。出力する音の数が多くなるほど一つの音の領域が狭くなり、望む領域に指先を合わせづらくなるため出力できる音高範囲を2オクターブまでとしている。これによって演奏中に範囲外の音を出力したい場合、出力範囲を変更する必要がある。しかし、音高範囲の変更を図3.1のジェスチャーを使って実装する場合、どのジェスチャーも認識精度が悪く演奏中に素早く動かす操作として扱いづらい。そこでジェスチャーに音高範囲の変更を割り当てるのではなく比較的素早く認識する深度を使って演奏モードと音高範囲の変更モードを切り替える方式にした。手をカメラ側に伸ばし音高範囲の変更モードにしたあと画面をスライドさせるような動きをすることで音高範囲が上下にスライドする。音高変更モードに切り替わる深度の範囲(以下、音高変更範囲)はユーザとカメラの距離で変わる。ユーザーの前方の一定範囲を常にtapを認識する範囲とし(以下、tap認識範囲)、tap認識範囲のカメラ側の端からカメラまでの範囲を音高変更範囲とした。図4.1に概要を示す。

## 4.2 実行結果と考察

図4.2、図4.3に実行結果を示す。音高範囲変更中であることがわかりやすいように指先が音高変更領域に入ったら色のついた円を表示させるようにした。出力範囲は赤、黄、緑、青、紫の順に低くなっていく。この手法によって出力範囲を変更すれば6オクターブ分の範囲の音を出すことができた。これは61鍵(5オクターブ)ピアノの音高範囲より多いのでユーザーが出したい音をカバーできると考えられる。しかし、音高範囲変更中の手は演奏ができないため演奏の自由度が下がるため、なるべく音高変更機能を使わないようにするため、音高範囲の音の数の多さとの一つの音の領域が操作しやすいサイズとなることを両立させた音高範囲の取り方を今後の課題としたい。

### 4.1. 実装手法



**①:音高変更範囲**

**②:tap認識範囲**

**③:指先**

図4.1: 音高変更範囲と tap 認識範囲

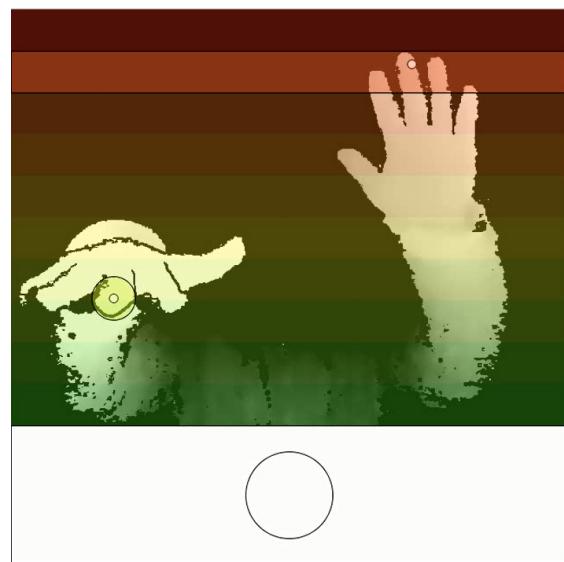


図4.2: 出力範囲の変更(高音)

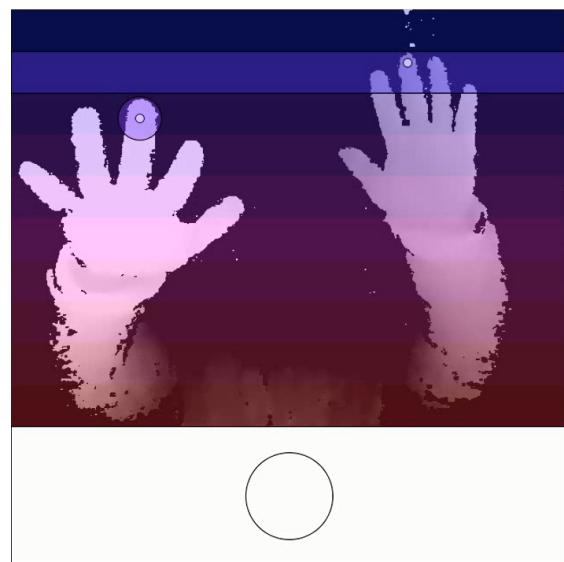


図4.3: 出力範囲の変更(低音)

#### 4.2. 実行結果と考察

## 第5章

### 調性制約の生成

本章では調性制約の生成方法について述べる

## 5.1 調性制約の形式

調性制約は、曲のコード進行に応じて出力可能な音の周波数列を変化させていく。本研究では調性制約は時間(ms), 出力可能な周波数列のcsvファイルとして生成され, 周波数列の選び方は(1)コードの構成音のみ, (2)コードの構成音+コードに対するメロディ頻度の高い音。の2種類の調性制約を生成した。

### 5.1.1 コードに対するメロディ頻度のカウント

Songleからのコードとメロディのデータ100曲分を使用してその統計をとる。まず、コードを表5.1に従って別のコードに分類する。表5.1は100曲分のコードで出現頻度の高いコードのみを載せている。分類後の各コードのときCからBまでの1オクターブ分12音が流れている時間を計測し、調の主音からの相対的な位置が等しいコードとメロディをまとめてカウントする。例えば、調の主音AのときのコードAでメロディG#が流れた場合と、調の主音CのときのコードCでメロディBが流れた時は同じ物としてカウントする。こうしてカウントしたメロディのながさの合計(ms)を各コードのメロディ出現頻度とした。メロディの周波数が1オクターブの範囲を超える場合は、オクターブ般化を行ってからカウントする。カウントは長調と短調は分けてカウントした。図5.1, 図5.2, 図5.3にそれぞれ長調のときのコードIM, コードIVM, コードVMのときの各メロディの出現頻度を、図5.4PFG Tonnetz上に表示したものをそれぞれ示す。図5.1, 図25.2, 図35.3を比べるとコードの根音が5度上にあがるとグラフの平面上でコードの三和音が右に平行移動したが、5度さがったとき単純な平行移動ではなく根音ではなく主音の頻度のほうが大きい。構成音以外の音の違いは根音の左上(6度)右上(7度)を比べると図5.1、図5.2は同じくらいで図5.3は右上がかなりすくない。図5.1と図5.4を比べると同じコードでもメロディ頻度の分布に違いが生じていて、短調のほうは原点の下側の領域に分布が出やすくなっている。

### 5.1.2 周波数列の決定

調性制約がもつ(2)の周波数列を決定する時に、コードに対するメロディの出現頻度から決める。コードの構成音に加えて、構成音以外でメロディ頻度が一番高いものを基準にその音に経験的に設定した倍率をかけた値と、コードに対するメロディの割合を閾値としてその閾値を上回った音を周波数列に加える。経験的に設定した倍率や閾値はコード構成音以外の音の頻度を降順にソートして平均をとつて設定した。

## 5.1. 調性制約の形式

表 5.1: コード変換表

変換前	変換後	変換前	変換後
XM	XM	Xsus4	XM
X-13	XM	Xadd2	XM
X-11	XM	Xadd4	XM
X-9	XM	Xadd9	XM
X-9	X7	Xaug	XM
X-5	XM	XaugM7	XM7
X-4	X7	Xdim	Xm
X6	X6	Xdim7	Xm
X7	X7	Xm	Xm
X9	X7	Xm11	Xm7
X11	X7	Xm13	Xm7
X13	X7	Xm6	Xm
X7-5	X7	Xm7	Xm7
X7-9	X7	XM7	XM7
X7-Aug	X7	Xm7-9	Xm7
X9-Aug	X7	Xm9	Xm7
X9-5	X7	XM9	XM7
X7+11	X7	Xmadd9	Xm7
X7+13	X7	XmM7	Xm
X7+9	X7	Xsus2	XM
X7b5	X7	Xsus4	XM

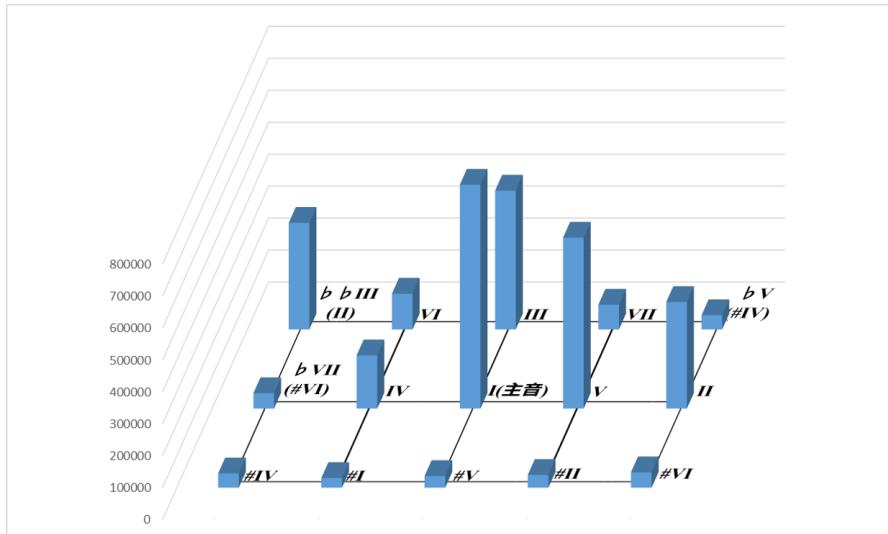


図 5.1: 長調のコード IM の各メロディの出現頻度

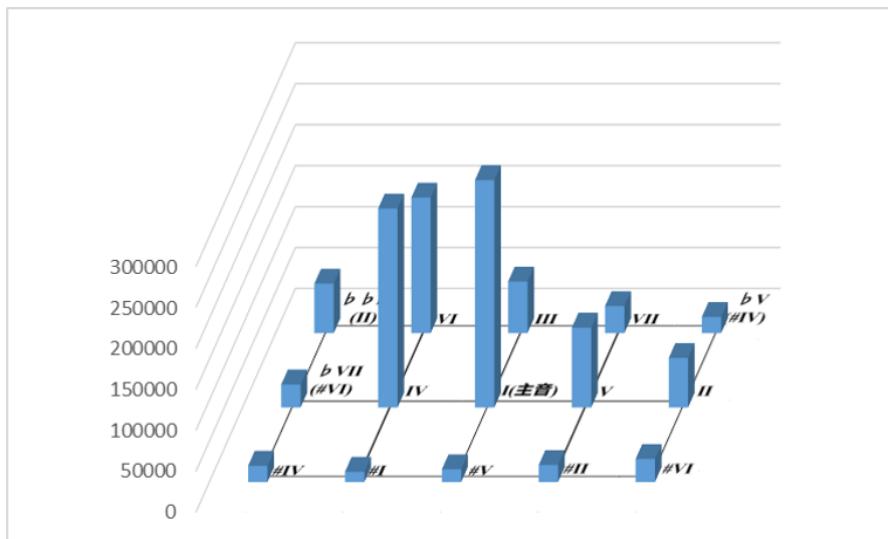


図 5.2: 長調のコード IVM の各メロディの出現頻度

## 5.1. 調性制約の形式

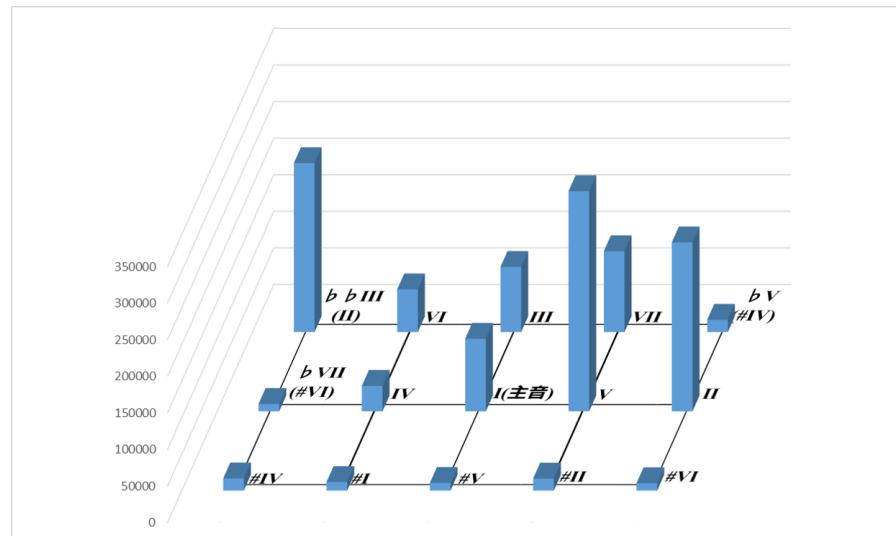


図 5.3: 長調のコード VM の各メロディの出現頻度

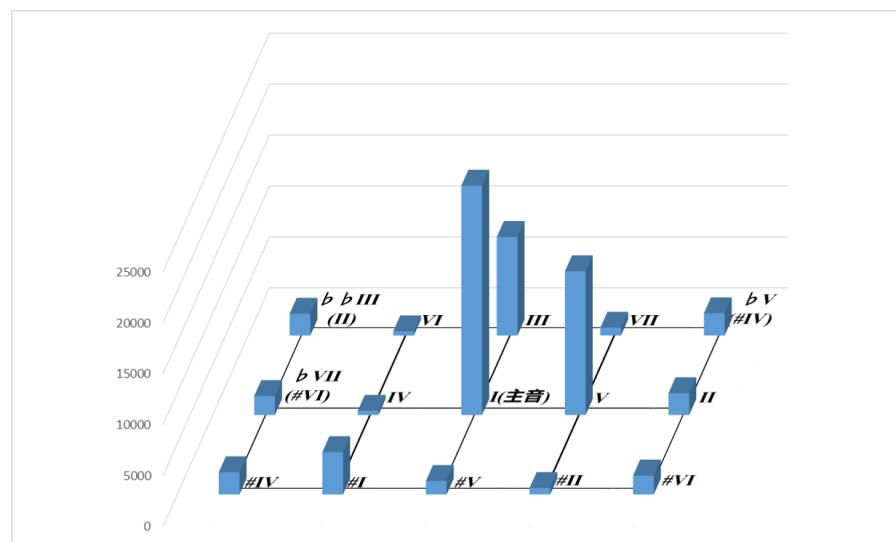


図 5.4: 短調のコード IM の各メロディの出現頻度



## 第6章

### 性能評価・考察

本研究の評価として、tap認識の精度と調性制約の評価実験を行った。

## 6.1 実験内容

20代から50代の男女12人に本システムを実際に体験してもらい、評価用紙の質問に答えてもらう。減衰音を出力するtap動作の認識手法については、第3章で説明した提案手法に対し、RealSense SDKで用意されたデフォルトの認識との比較を行った。調性制約の生成手法については、第5章で説明した提案手法に対し、単純に背景楽曲のコードの構成音を調性制約として用いる手法との比較を行った。ただし、提案手法と比較手法の順序が固定されると被験者による操作習熟の影響が生じるため、半分の6人は提案手法が先で比較手法が後、もう半分の6人は比較手法が先で提案手法が後になるように、実験を行った。

### 6.1.1 tap評価実験

第3章で説明したtap認識の提案手法とRealSense SDKで用意されたデフォルトのtap認識の2つの方法で、背景楽曲の拍を表す円に合わせてtapをしてもらい、各実験の後図6.1の二つの質問に答えてもらった。

### 6.1.2 調性制約評価実験

コードの構成音のみを周波数列にもつ調性制約と、Songleのデータから統計的に生成した調性制約で演奏してもらい、各演奏の後図6.2の2つの質問に答えてもらった。

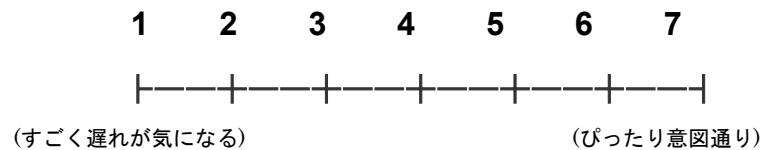
## 6.2 実験結果と考察

tap評価実験の結果の平均を図6.3に示す。調性制約評価実験の結果の平均を図6.4に示す。図6.3より、本研究で実装したtap認識の方がRealSense SDKのtap認識よりも高い評価を得ているので、実用的であると言える。図6.4より統計的に生成した調性制約は、コードの構成音のみを周波数列にもつ調性制約と比べ不協和音が含まれるが、わずかに意図通りの演奏がしやすいという評価を得た。これはコードの構成音にさらに音を追加することで表現の幅が広がったためと思われる。不協和音が多いため、追加する音を決める閾値の調節または追加する手法の検討を今後の課題としたい。

### 6.1. 実験内容

tap 評価実験

1. 音が出た時、それは意図通りのタイミングでしたか？



2. tap が認識されず音が出ない時がありましたか？

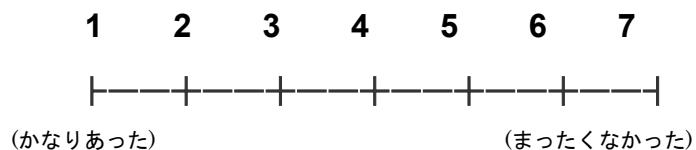
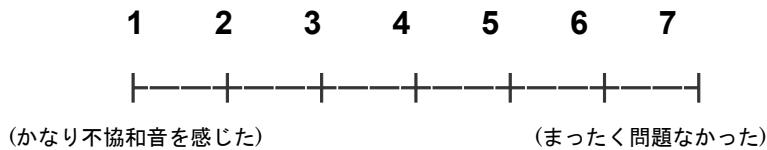


図 6.1: tap 評価用紙

## 6.2. 実験結果と考察

## 調性制約評価実験

1. 不協和音がなっていたと感じましたか？



2. あなたの演奏の意図どおり音が上がり下がりしていましたか？



図 6.2: 調性制約評価用紙

### 6.2. 実験結果と考察

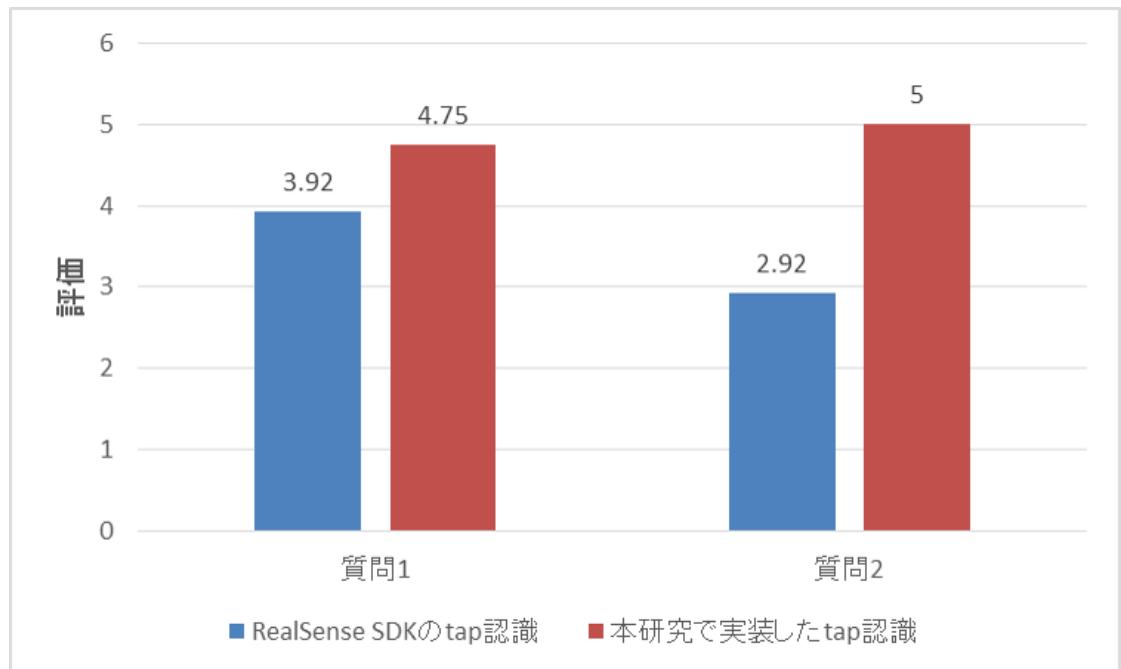


図 6.3: tap 評価実験結果

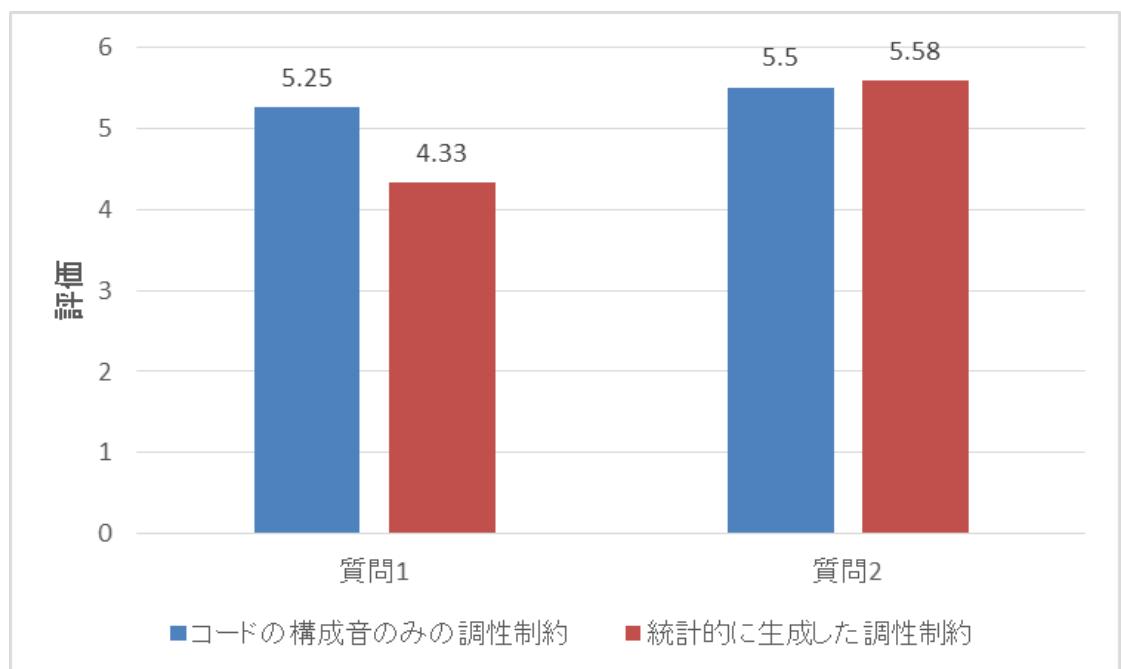


図 6.4: 調性制約評価実験結果

## 6.2. 実験結果と考察



## 第7章

### まとめ

## 7.1 本研究の要約

本研究では直感的にリズムと旋律線を入力すれば背景楽曲との調性を満たす音を出力する演奏インターフェースを実現した。リズムと旋律線は RealSense カメラを利用して入力し、ジェスチャー認識機能によってインターフェースの操作を行う。RealSense SDK のジェスチャー認識機能には認識漏れや、遅延などが目立ったため、本研究では指先や手のひらの速度や移動距離の閾値、指の開閉度などのパラメータを利用してジェスチャーの認識精度を向上させた。また、深度と身体動作から演奏中に発音可能な音高の範囲を指定できる機能を追加して演奏の自由度をあげた。出力される音はあらかじめ Songle から統計的に生成した背景楽曲との調性制約によって決定し、発音可能な周波数列をコード進行に応じて変化させる。特に認識漏れや遅延の多い tap 認識の改善と生成した調性制約について、評価実験を行ったところ、改善した tap 認識機能は RealSense SDK と比べ実用的な精度であり、調性制約はコードの構成音のみを周波数列とする調性制約より不協和音が多いがわずかに意図通りの演奏がしやすいという評価を得た。

## 7.2 今後の展望

今後の課題・展望としては機能面において、音量変更機能や、楽器音の出力機能の実装や調性制約の精度向上。マウス、タブレットなどを入力装置とした本システムの実装。ハッカソンなどで簡単に扱えるよう API 化など、他の入力装置への対応や連携の充実化を図りたい。将来的には、地域社会振興のために開催される音楽イベント、アーティストのコンサートなど多人数で同時に即興合奏ができるように、複数人の身体動作入力の処理、画面を見ないでも操作できるような直感的な操作、オンラインで離れた人と合奏などを今後の課題としたい。

### 7.1. 本研究の要約

## 謝辞

本論文を執筆するにあたり、多くの方のご支援ご協力を賜りました。この場で皆様に感謝の言葉を申し上げたいと思います。

まず、名古屋工業大学大学院 工学研究科 情報工学専攻の白松 俊准教授に厚く御礼申し上げます。白松教授には日々の研究活動に限らず、音楽知識の乏しい私に懇切丁寧にご指導して頂き、数多くの貴重な体験もさせて頂きました。ここに心より感謝致します。

同研究室の同輩である成瀬 雅人君、西田 拓哉君、山野 太靖君とは特に苦楽を共にしてきました。その中で、皆の頑張り、励まし、助言に助けられることも多く、充実した日々を過ごすことができました。ここでみなさんに感謝の意を表しつつ、今後の活躍をお祈り申し上げます。

その他、バイト先のとだがわこどもランドの皆様をはじめ、多数の友人たちにも評価実験の協力を頂きました。ここで合わせて御礼申し上げます。

最後に、研究活動で帰りが遅くなることもあり迷惑をかけた私を、いつもと変わらず支えてくれた家族に、心より感謝致します。



## 参考文献

- [1] 波多野謙余夫(編):”音楽と認知” Vol. 8, 東京大学出版会, (2007)  
 様々な認知研究や音楽理論から音楽心理学を, 人間はいかにして音楽を認知するかの研究として捉えて考察をのべている.
- [2] 能動的音楽鑑賞サービス Songle,<http://songle.jp/>, 2015年11月25日アクセス.  
 能動的音楽鑑賞サービス Songle は動画サイトやネット上にアップロードされた合法的に視聴できる音楽を自動解析し, 音楽と一緒にサビ, メロディ, コード, ビートを視覚的に表示させて鑑賞することができる.
- [3] 音楽之友社:”音楽中辞典” 音楽之友社, (1979)  
 音楽用語の解説・用法を記載した辞典.
- [4] 菅道子. ”身体表現を取り入れた参加型音楽コンサートの可能性: カノンの理解を目指した「追いかけっこしよう」の事例から” 和歌山大学教育学部教育実践総合センター紀要 18 , pp. 121-129, (2008)  
 音楽理解における身体表現の有効性を小学校低学年を対象とした参加型音楽コンサートの企画・実施を通して検討し, 旋律線のてなぞりなどの身体動作が音楽理解を促進することをあげている.
- [5] 高橋祐樹:”私の研究開発ツール Processing” 映像情報メディア学会誌 Vol. 64, No.12, pp. 1841-1849, (2010)  
 Processing の説明と, 基本的な使い方について述べている.
- [6] 中村俊介, 竹井将紫: ”インタラクティブアート 「KAGURA」 によるワークショップ: 東京ミッドタウン・デザインハブ・キッズウィークにおける子供向けイベントの報告 (B-2 音楽と聴覚のデザイン, 研究発表, 芸術工学会 2010 年度秋期大会 in 浜松).” 芸術工学会誌, Vol. 54, pp. 48-49, (2010)

カメラの前で動いて音声を生成するインタラクティブアート「KAGURA」をデジタルでしかできないインタラクティブな教育コンテンツとして利用しようと考え、東京ミッドタウンで開催された小・中学生向けワークショップで展示したときの報告が書かれている。

[7] KAGURA,<http://www.kagura.cc/jp/>,2015年11月25日アクセス

Real Senseに対応した身体動作とジェスチャーで操作できる音楽演奏アプリケーション。

[8] 菅家浩之, 竹川佳成, 寺田努, 塚本昌彦:”Airstic Drum: 実ドラムと仮想ドラムを統合するためのドラムステイックの構築” 情報処理学会論文誌, Vol. 54, No. 4, pp. 1393-1401, (2013)

楽器の運搬と演奏スペースの問題を解決するため、使用頻度の低い打楽器に仮想的なドラムを割り当て加速度センサの閾値から仮想ドラムの叩打と実ドラムの叩打を区別する方法について述べている。

[9] Shiramatsu, S., Ozono, T., and Shintani S.: A Computational Model of Tonality Cognition Based on Prime Factor Representation of Frequency Ratios and Its Application. Proc. of SMC 2015, (2015)

調性理解モデル PFG Tonnetzについて述べている。

[10] Behringer, R. and Elliot, J.: Linking Phys- ical Space with the Riemann Tonnetz for Exploration of Western Tonality, chapter 6, pp. 131143, Nova Science Publishers, (2010)

1880年にRiemannが発展させたTonnetzを分析し、数式化した。

[11] Intel RealSense SDK 2015 R5 Documentation, [https://software.intel.com/sites/landingpage/realsense/camera-sdk/v1.1/documentation/html/index.html?jointtype\\_pxchanddata.html](https://software.intel.com/sites/landingpage/realsense/camera-sdk/v1.1/documentation/html/index.html?jointtype_pxchanddata.html), 2016年1月25日アクセス

RealSenseが認識できる関節の種類が載っている。

## 付録A

# 平成27年度 電気・電子・情報関係学会 東海支部連合大会

身体動作による旋律線入力に基づく  
音楽未経験者でも即興合奏可能な演奏インターフェースの試作  
一ノ瀬 修吾\*, 白松 俊(名古屋工業大学)

Implementing an interface enabling music beginners to play improvised ensemble based on inputting pitch contour by body movement

Shugo Ichinose, Shun Shiramatsu (Nagoya Institute of Technology)

### 1. はじめに

旋律歌唱や旋律聴取における 3 つの処理側面として、リズム、旋律線、調性がある[1]。このうち、音楽未経験者の即興合奏を難しくするのは調性であり、リズムや旋律線については、比較的容易に直感的な動作として表出できる。なお旋律線は旋律概形とも呼ばれ、大まかな音高の上下動を指す。本研究では、ユーザーが旋律線とリズムを身体動作で入力すれば、システムの音高補正により調性的制約を満たす合奏ができる演奏インターフェースの実現を目指す。調性的制約については我々が過去の研究で提案した PFG Tonnetz (Prime Factor-based Generalized Tonnetz) [2]を用い、本稿では特に身体動作による演奏インターフェースに焦点を当てる。

身体動作による演奏インターフェースでは、(1)直感的な身体動作による旋律線の入力手法と、(2)減衰音と持続音の区別やオフセットタイミングを誤認識無く指定できる入力手法が課題となる。(1)の直感的な入力手法としては、Intel RealSense 3D Camera (以下、RealSense) を用い、手の動きを認識して音高の上下動 (すなわち旋律線) を入力する。RealSense は手指のジェスチャー認識が容易であり、(2)の減衰音と持続音の区別やオフセットタイミングを手指のジェスチャーで入力するのに適している。

### 2. 実装方法

Fig.1 にシステム構成図を表す。本システムの実装には、RealSense に対応し、音声処理にも適した Processing を用いる。まず、RealSense がユーザーの手指の情報をジェスチャーを検出する。手の高さの変化が旋律線入力となり、ジェスチャーが持続音／減衰音の区別やオフセットタイミングの入力となる。あらかじめ与えられた背景楽曲に対し、時刻ごとに取りうる音高の調性制約を PFG Tonnetz [2] に基づき算出しておく。これを用い、入力された手の高さと近く制約を満たす音高を決定する。これにより、背景楽曲に対して不協和とならないサイン波を生成する。

持続音／減衰音の区別については、タップ動作 (手をカメラに向かながら前後に動かす) が減衰音の指定となり、サムアップ動作 (親指を上に立てる) が持続音の指定となる。また、フィスト動作 (手指をすべて閉じる) で持続音を止めることができる。これらのジェスチャーは、どれも手の形状が異なり誤認識が起こりにくいため、自在に減衰音と持続音を区別し出力できる。左右の手で出力される演奏音は独立して



Fig. 1 System Architecture and User Interface

おり、2 和音まで出力可能である。

調性制約の計算に用いる PFG Tonnetz[2]は、音の周波数比を素数指数表現をした上で指数を平面上にプロットし、その整数格子点で協和音程を表すモデルである。これにより算出された時刻  $t$  の調性制約を  $c(t)$ 、手の高さを  $x(t)$  とすると、Fig. 1 の音高決定器は調性制約を満たす周波数  $f(t)$  を以下のように決定する。

$$f(t) = \arg \min_{f \in c(t)} (f - f'(t))$$

$$f'(t) = f_{\text{tonic}} \cdot \exp(\alpha(x(t) - x_{\text{tonic}}))$$

なお、 $f_{\text{tonic}}$  は調の主音の周波数、 $x_{\text{tonic}}$  は主音に対応する手の高さ、 $\alpha$  は手の高さと音高の変化率の調性パラメータである。

### 3. おわりに

本稿では、音楽未経験者でも不協和音を出さず合奏できる演奏インターフェースを試作した。今後は、被験者実験によってユーザーの意図どおりの合奏が可能か検証する必要がある。また、タップ動作による減衰音指定時に、ジェスチャーの速度による音量変更機能を追加する予定である。

### 文献

[1] 波多野 誠余夫: 音楽と認知、認知科学選書、第 12 卷、東京大学出版会 (1987)

[2] Shiramatsu, S., Ozono, T., and Shintani S.: A Computational Model of Tonality Cognition Based on Prime Factor Representation of Frequency Ratios and Its Application. *Proc. of SMC 2015* (2015) (to appear)

## 付 錄B

**IPSJ2016  
第78回情報処理全国大会(掲載予定)**

## 調性判断の不要な身体動作入力による 即興合奏支援システムの試作

一ノ瀬 修吾<sup>†</sup> 白松 俊<sup>‡</sup>

名古屋工業大学 工学部情報工学科<sup>†</sup> 名古屋工業大学 大学院工学研究科情報工学専攻<sup>‡</sup>

### 1. はじめに

旋律歌唱や旋律聴取における 3 つの処理側面として、リズム、旋律線、調性がある[1]。旋律線 (pitch contour) は旋律概形とも呼ばれ、大まかな音高の上下動を指す。調性は、調やコード進行を指す。これら 3 要素のうち、音楽未経験者の即興合奏を難しくするのは調性判断である。調性判断とは調の種類を判定することを指す。調性以外の旋律線やリズムについては、初心者でも比較的容易に直感的動作として表出できる。そのため、例えば地域振興のために幅広い層の自由な参加を志向した音楽イベントであっても、音楽未経験者は手拍子など自由度が低い手段での参加に限られる。

そこで本研究では、ユーザの調性判断をシステムが補うことで、音楽経験に乏しい人でも調性感を損なわず自在に即興合奏に参加できるようなシステムの開発を目指す。具体的には、ユーザが旋律線とリズムを身体動作で入力すれば、システムが背景楽曲のコード進行に応じて音高補正を行い、調性の制約を満たし不協和にならない合奏ができる演奏インターフェースの実現を目指す。そのために、以下の 3 つの課題を扱う。

- (1) 直感的な身体動作による旋律線やタイミング、音の種類などの入力手法
  - (2) 演奏音の出力範囲の指定手法
  - (3) 背景楽曲のコード進行に対する調性の制約の決定手法
- (1)の直感的な入力手法としては、Intel RealSense 3D Camera (以下、RealSense) を用い、手の動きを認識して旋律線を入力する。RealSense は手指のジェスチャー認識が容易であり、直感的な身体動作入力に適している。
- (2)の演奏音の出力範囲の指定手法は、手を伸ばし画面をスクロールさせるような操作モード

**Implementing an Improvisational Ensemble Support System Based on User's Body Motion with Redeming Tonality Recognition**  
Shugo Ichinose<sup>†</sup>, Shun Shiramatsu<sup>‡</sup>  
<sup>†</sup>Department of Computer Science, Faculty of Engineering,  
Nagoya Institute of Technology  
<sup>‡</sup>Department of Computer Science and Engineering, Graduate  
School of Engineering, Nagoya Institute of Technology

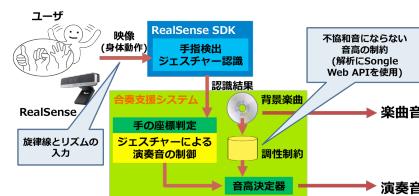


図 1: システム構成図

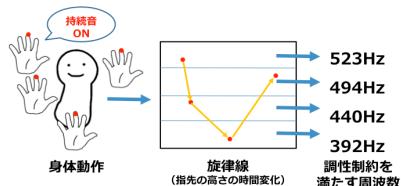


図 2: 直感的な身体動作による旋律線の入力を設け、手の深度によってこの操作モードを切り替える。

(3)の調性制約の決定手法は、能動的音楽鑑賞サービス Songle[2]の Web API を用いて実現する。

### 2. 実装方法

図 1 にシステム構成図を示す。本システムの実装には、RealSense に対応し、音声処理にも適したプログラミング言語 Processing を用いる。まず、RealSense がユーザの手指の情報とジェスチャーを検出する。図 2 に直感的な身体動作による旋律線の入力の概要を示す。手の高さの時間的変化が旋律線入力となり、ジェスチャーが持続音／減衰音の区別やオフセットタイミングの入力となる。画面上にあらかじめ与えられた背景楽曲のコード進行に応じて変化する領域を設け、指先がこれに触ることで、それぞれの領域に対応した背景楽曲に対して不協和とならないサイン波を出力する。

持続音／減衰音の区別については、タップ動作（手をカメラに向かながら前後に動かす）が持続音の指定となり、サムアップ動作（親指を

上に立てる)が持続音の指定となる。また、ファイスト動作(手指をすべて閉じる)で持続音を止めることができる。左右の手で出力される演奏音は独立しており、2和音まで出力可能である。また、持続音を出力する際に現在演奏している音の領域と離れた領域の音を出力する場合、手を移動させた時に通った領域の音まで出力してしまう。これを防ぐため、Songle Web API から取得した背景楽曲の拍に合わせて点滅する円を画面に表示させ、リズムを取りやすくするとともに、円が表示されてないタイミングのとき演奏音をボルタメントさせることで、離れた領域の音を出力するときに間の領域の音を出力するのを防ぐ。

画面上に存在できる出力領域には制限がある。そのため、表示された出力領域よりも高い(低い)音を出力したい場合のための操作モードを設ける。RealSense は手の深度を検出することができるため、手の深度に閾値を用意し、これを超えた場合出力画面の操作モードに切り替わる。このモードでは、出力領域をスクロールさせると出力領域が移動し、より高い(低い)出力領域を表示させることができる。

調性制約は Songle Web API の自動解析を元に作成する。100 曲分の曲のコード情報とメロディ情報を取得し、各コードの構成音以外の発生頻度の高い音を調べる。その分析には我々が提案する調性の数理モデル[3]を用いる。背景楽曲の各コードの構成音と頻出音の周波数列を(ミリ秒、周波数列)の csv ファイルとして出力したものを背景楽曲の調性制約として使用する。

### 3. 実行結果と考察

図 3 にシステムの実行結果を示す。図 3 の①の小さい円は認識された手指の座標、②は拍、③は出力領域操作モード中を示す円である。

動作品質に関しては以下の 5 つの点を満たすものが望ましいと考えられる。

- (1) ジェスチャーの認識から演奏音の出力までのタイムラグが演奏者の違和感にならない程度のものであること。
- (2) 出力領域のスクロールは、なるべく演奏の妨げにならないよう最小限の動作でできること。
- (3) 演奏したい音を誤認識なく出力できること。
- (4) 児童、高齢者などが扱えるように難しい動作を必要としないこと。
- (5) 音楽未経験ユーザでも意図どおりに合奏できていると感じられること。

本システムの場合、(1)はタップの認識から演

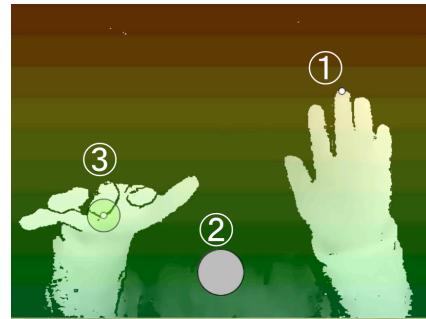


図 3 : システムの実行画面

奏音の出力までに遅延や認識漏れがあり改善が必要である。(2)は、出力領域をスクロールする際、領域を移動させている方の手では、持続音は音を変更することができず減衰音は出力することができなかった。(3)について出力領域は、画面を現在時刻の周波数列の数で分割した領域であり、出力する周波数の数が多いと出力領域は狭まり、はやく手を動かした場合間違った音を出力することがある。しかし周波数の数を減らすと出力領域を変えたいと感じる頻度が増え、演奏の妨げとなる動作が増えることにつながるために調整が必要である。(4)難しい動作は必要とせず、拍が表示されるため、リズムに合わせて旋律線を入力しやすいため初めて本システムに触れる人でも扱いやすい。(5)は今後、被験者実験による検証が必要である。

### 4. おわりに

本研究では直感的に旋律線とリズムを入力すれば背景楽曲との調性を満たす音を出力する演奏インターフェースを実現した。今後の課題としては、被験者実験、多人数による同時即興合奏、PCM 音源などの楽器音対応、音量変更機能が挙げられる。

**謝辞** 本研究は JSPS 科研費(25870321)の支援を受けた。

### 参考文献

- [1] 波多野 誠余夫(1987). 音楽と認知、認知科学選書、第12巻、東京大学出版会。
- [2] Goto, M., Yoshii, K., Fujihara, H., Mauch, M., & Nakano, T (2011). Songle: A Web Service for Active Music Listening Improved by User Contributions. *Proc. of ISMIR 2011*, pp. 311-316.
- [3] Shiramatsu, S., Ozono, T., and Shintani S. (2015). A Computational Model of Tonality Cognition Based on Prime Factor Representation of Frequency Ratios and Its Application. *Proc. of SMC 2015*, pp. 133-139.



# 卒業論文

(題 目)

MissionForest: 組織内外における  
協働支援のためのタスク構造化システムの試作

指導教員 白松 俊 准教授

名古屋工業大学  
工学部 情報工学科

平成 23 年 4 月 入学

(学籍番号) 24115054  
(氏名) 後藤 誉昌

(提出日: 平成 28 年 2 月 8 日)