

研究報告

複数の種類のセンサを利用した音楽演奏システム“HUMOSIC”について A STUDY ON THE EMPLOYMENT OF DIFFERENT TYPES OF SENSORS IN THE MUSIC PERFORMANCE SYSTEM “HUMOSIC”

吉田紘希

Hiroki Yoshida

九州大学大学院芸術工学府

Graduate School of Design, Kyushu University

中村滋延

Shigenobu Nakamura

九州大学大学院芸術工学研究院

Faculty of Design, Kyushu University

概要

現在、Kinect や Leap Motion を始めとした様々なセンサを用いた音楽演奏システム及びメディアアートが制作されているが、それらの多くは一種類のセンサのみを用いたものである。そこで本作品《humosic》では、センサの中でも特にモーションセンサを複数種類用いることにより、筆者が制作した Kinect を使用した音楽演奏システム《.tial sampler》の表現をさらに拡張することを目的としている。

Today, various types of sensors (e.g., Kinect or Leap Motion) are used in many music performance systems and media arts projects; however, most use only one type of sensor. In "humosic," a music performance system, I used different types of sensors (especially motion sensors) to expand the expressive potential of ".tial sampler," a performance system I had devised in the past, which used Kinect.

1.2. 背景

近年、クリエイティブ・コーディング [1] による作品に Kinect[2] や Leap Motion¹ といったモーションセンサを取り入れるケースが多く見受けられる。これらのセンサ及びセンサを使用する際に必要となるライブラリの普及により、ナチュラル・ユーザ・インターフェースとなるシステムを安価に設計することが可能となり、メディアアート作品にも多く取り入れられるようになった。モーションセンサが利用されているメディアアートの中でも特に音表現に関わる代表的な例としては、The V Motion Project[3] が挙げられる。The V Motion Project では、演奏者の体の動きを Kinect によってパソコン用コンピュータ(以降 PC)に取り込み「特定の音に応じた複数の仮想的なセンサに触れる」や「手などの特定の部位の相対的な距離によって音色をコントロールする」といった演奏方法をシステムに取り入れている。それにより、演奏者は楽器などに触れることなく音楽を演奏することが可能となっている(図 1)，



図 1. The V Motion Project

1.1. 目的

本稿は筆者の意図する音楽演奏を行うために制作した、メディアアート表現としての音楽演奏システム《humosic》(ヒューモジック)について、それまでに作成した同種のシステム《.tial sampler》(スペーシャルサンプラー) や《.musician》(ドットミュージシャン), 《Color Tracker》を踏まえて解説し、さらに今後製作する作品の可能性を考察するものである。

一方で楽器においては、演奏者が楽器に触れず演奏で

¹ <https://www.leapmotion.com>

きるものとしてテルミン [4] が挙げられる、テルミンは演奏者が空中に掲げる手の位置によって音高と音程を調節することができる楽器であり、手の動きによって連続的に変化する音高や音量をコントロールすることでヴァイオリンのような音色を奏でることができる。

これらのことから筆者の制作した《humosic》では、The V Motion Project やテルミンのように演奏者が楽器と接触することなく、体の動きによって音楽を演奏できる演奏システムを設計することを指針とした。またこの指針を踏まえ、システムを設計する上で重きを置いたことは、センサとなる部分に触れる/触れないといったような「0と1の離散的な変化」と「体の動きに伴う連続的な変化」を融合させることであり、それを実現させるために複数のモーションセンサを取り入れるという方法を用いた。

2. これまでの作品

今回制作した《humosic》及び、それまでに製作した作品と共に通する「触れることなく演奏できる演奏システムの設計」という目的が根幹にあり、その目的を達成すべく制作した演奏システムについて順を追って説明する。

2.1. 《Color Tracker》

《Color Tracker》は、PC に内蔵された（もしくは PC に接続された）RGB カメラから得られた映像を分析し、画面上の赤、緑、青の 3 色の位置によって音をコントロールする作品である。この《Color Tracker》は Max² パッチとして構成されており、映像の処理及び音の生成は全て Max 上で行われている。

演奏者は RGB カメラに対して上記 3 色の「もの」を提示することにより、各色に対応する音を生成することができ、さらに「もの」の位置を変更することによりそれぞれの音色や音量、音程を操作することができる。

図 2 のように、RGB カメラの映像上で認識された「もの」の周りに各色の長方形が追加して表示されている。長方形が大きくなるほど色に対応した音が大きくなり、長方形の位置が右になるほど音程が高く、上になるほど明確な音色になる。

《Color Tracker》は、簡単かつ直感的に音を操作できるよう設計しているため、正常に動作すれば音が連続的に変化するはずであったが、RGB カメラによる色の認識は照明など周囲の明るさに左右されやすいことから意図しない挙動を見せることが多々あり、RGB カメラによる演奏システムの設計には限界があるということが判明した。

² <http://cycling74.com/products/max/>

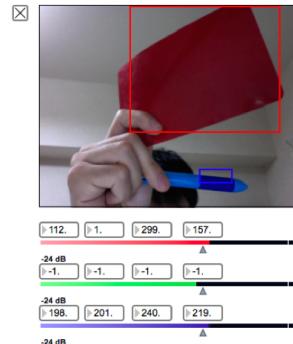


図 2. 《Color Tracker》の演奏画面

2.2. 《.musician》

《.musician》はサンプラーのような楽器を現実空間に拡張した演奏システムを目標としており、RGB カメラに加え深度センサも内蔵している Kinect を使用した。この深度センサを主に使うことにより、《Color Tracker》で問題となった RGB カメラの不安定さを解消し、さらに、音を連続的に再生、変化する《Color Tracker》に対して、演奏画面上に仮想的なセンサを配置することで音の再生や停止という離散的な変化を演奏者がコントロールできるように設計した。

この作品も Max パッチとして制作しており、演奏者は演奏画面上に表示される 9 色の長方形に白い影が映るように手を押し出すなどの動作を行うことにより、各々の色に対応した音源（事前に収録したもの）を再生することができる。また、長方形の中でも位置によって音源の再生速度をコントロールすることができ、さらに各音源の再生位置を演奏画面によって視覚的に確認することができる（図 3）。

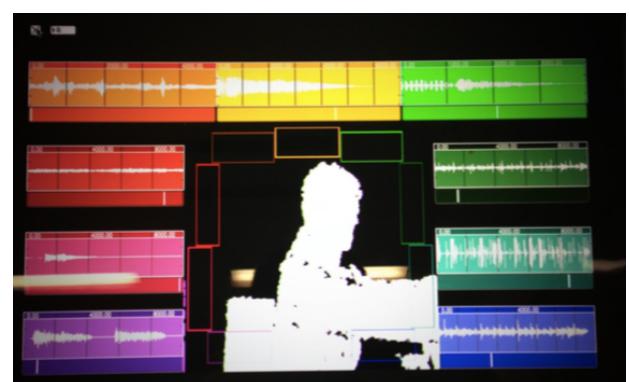


図 3. 《.musician》の演奏画面

先にも述べたが《.musician》には Kinect の深度センサを主に使用しており、この深度センサは赤外線マッピングと赤外線カメラを用いているため、室内での使用であれば照明の変化などに影響されず暗所でも演奏するこ

とが可能となっている。しかし一方で《.musician》では深度センサによる Kinect と演奏者の距離のみの情報を使用しているため、人間の「動き」に直接対応した連続的な表現をするには至らなかつたという課題が生じた。

2.3. 《.tial sampler》

《.tial sampler》もそれまでの作品と同様に Max パッチとして制作しているが、《.musician》で生じた課題を解決すべく、ナチュラルインターフェイス向けのライブラリである OpenNI の中でも特に人体の姿勢認識を担う NITE を使用した OSCeleton³ を追加することにより、演奏者の動きを三次元骨格位置情報に変換し OSC⁴ 通信することが可能となった。

演奏画面は《.musician》に類似しているが、演奏者は片方の手で音源に対応する長方形に触れ、同時に反対の手を前に突き出すことによって、音源を演奏者の意図する再生速度⁵で再生することができる。音源の再生速度及び音源の停止は、長方形に触れる手の位置によってコントロールすることができ、もう一方の手は再生停止を決定するスイッチのような役割を担うことになる。各音源の再生位置や再生速度は《.musician》に比べ演奏画面上でも視認しやすいうように改良している（図 4）。

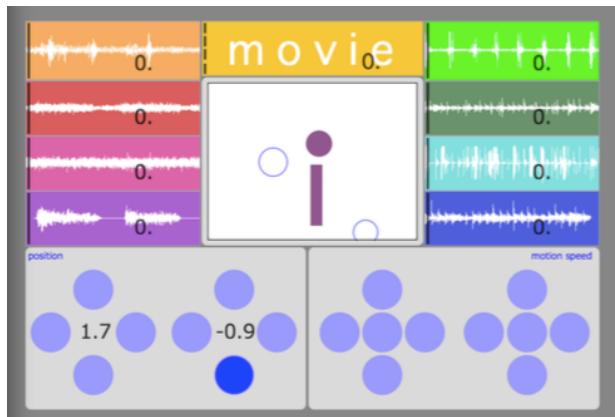


図 4. 《.tial sampler》の演奏画面

《.tial sampler》では、これらのような演奏システムの改善に加え、再生される音や演奏者の動きに応じてリアルタイムに変化する視覚的イメージも生成し、聴取者に対してはこの視覚的イメージを提示した（図 5）。

《Color Tracker》が連続的な音の変化しか扱えなかつたことに対し、《.tial sampler》では両手を別々に認識することにより《.musician》で可能にした離散的な変化をより正確なものにできた。しかし、そのために演奏方法

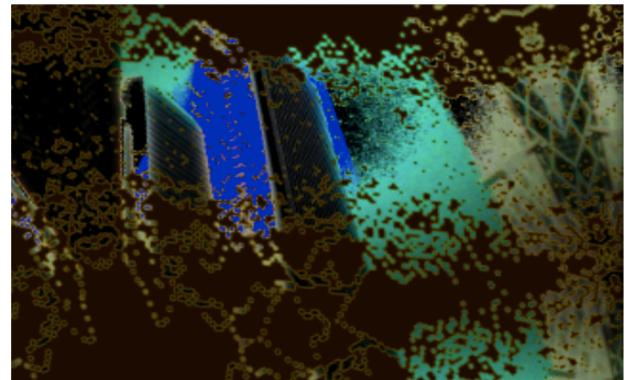


図 5. 《.tial sampler》の視覚的イメージ

が難し難しくなってしまったことに加え、演奏者動きに対応した連続的な操作を行うことができる状態ではなかつた。

3. 《HUMOSIC》

上記 3 つの先行作品を受け《humosic》では、より演奏者が意図した操作を正確且つ直感的に行うという目標を掲げ、その解決策として複数のモーションセンサを使用して演奏者の「動き」の認識をさらに拡張することとした。

3.1. システム概要

先行作品と異なり、《humosic》は基本的なシステムに openFrameworks[5] を、音響生成に関わるシステムに Pure Data[6] を使用して制作した。また、モーションセンサとして Kinect と Leap Motion を同時に使用しており、これら全てを openFrameworks で統合して制作するために以下のアドオン⁶を主に使用した。

- 1) ofxPd[7]
- 2) ofxOpenNI[8]
- 3) ofxLeapMotion[9]
- 4) ofxOpenCV

3.2. ユーザインターフェース

本作品を用いて演奏を行う際、演奏者には図 6 の画面が提示される。音を発生するための基本的な仕組みは《.tial sampler》に似ているが、《humosic》では音の再生・停止を片手のみで行うように変更した。つまり、演奏者は音に対応付けられた複数の仮想的なセンサに片方の手で触ることで発音でき、自由になったもう一方の手でさらに他の音を発音できる他、手を上下左右に

³ <https://github.com/Sensebloom/OSCEleton/>

⁴ OpenSound Control, 音楽演奏データをネットワーク経由でリアルタイムに共有するための通信プロトコル。

⁵ 逆再生を含む

⁶ サードパーティ製のライブラリやフレームワークの総称、openFrameworks の機能を用途に合わせて拡張できる、

動かすことで発音される音の音高や音量を連続的に変化させることができる。さらに、仮想的なセンサに触れる際の指の状態により簡易的に和音を発音することも可能である。これにより、演奏者は意図した音を直感的に演奏・操作することが可能となる。また生成される音は、《trial sampler》のように事前に収録したものではなく Pure Data によって生成されたシンセサイザーのような合成音声である。

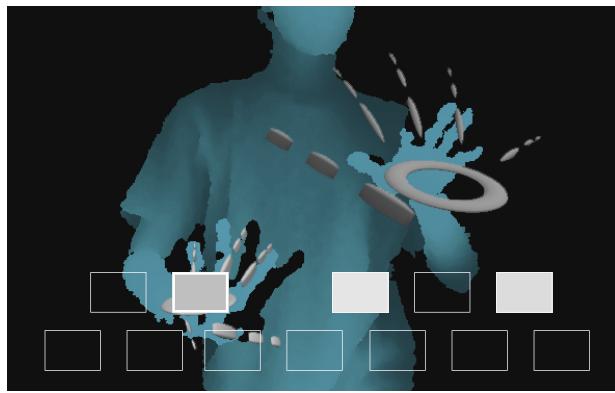


図 6. 《humosic》の視覚的イメージ

4. 考察と今後の展望

《humosic》について、現時点での問題点とそれに対する今後の展望をまとめに代える。

4.1. ユーザインターフェース

《humosic》ではそれまでの作品とは異なり、音の離散的な変化と連続的な変化を直感的に操作することが可能になった。しかし一方で、演奏者に提示されるユーザインターフェイスがシンプルすぎるために演奏への没入感が十分に得られないという問題が生じた。これに対して、音の変化によって背景を変化させるなど、演奏者に提示する映像そのものを改良することはもちろん、映像を投影する媒体もについても検討を行いたいと考えている。

4.2. モーションセンサ

《humosic》では Kinect と Leap Motion をモーションセンサとして使用したが、どちらも赤外線マッピングを利用しているため設置方法によっては得られる情報にノイズが発生しやすくなる。これに対して、今後は筋電センサをベースとしたウェアラブルセンサーの Myo⁷など仕組みの異なるセンサを組みあわせることによってこの問題を解決したいと考えている。

⁷ <https://www.thalmic.com>

4.3. パフォーマンスツールとしての可能性

《humosic》は演奏者が体を動かすことによって音楽を演奏するシステムであるため、演奏者の体の動きそのものにも表現としての面白さを付加できるよう改良することで、パフォーマンスツールとしての可能性も模索していきたい。

5. 参考文献

- [1] 田所淳 “クリエイティブ・ミュージック・コーディング” オーディオ・ビジュアル作品のためのオープンソースなソフトウェア・フレームワークの現状と展望”, 先端芸術音楽創作学会 会報 Vol.5 No.4, pp15-20, 2009.
- [2] 西林孝 小野憲史 “キネクトハッカーズマニュアル”, 株式会社ラトルズ, 2011.
- [3] The V Motion Project, <http://www.custom-logic.com/blog/v-motion-project-the-instrument/>.
- [4] Theremin World, <http://www.thereminworld.com>.
- [5] openFrameworks, <http://openframeworks.cc>.
- [6] Pure Data, <http://puredata.info>.
- [7] ofxPd, <https://github.com/danomatika/ofxPd>.
- [8] ofxOpenNI, <https://github.com/gameoverhack/ofxOpenNI>.
- [9] ofxLeapMotion, <https://github.com/Theo/ofxLeapMotion>.

6. 著者プロフィール

吉田紘希 (Hiroki Yoshida)

1989 年福岡県生まれ。九州大学芸術工学部音響設計学科卒業。九州大学大学院芸術工学府の修士課程に在学中。クリエイティブ・コーディングによる音楽系メディアアート作品の中でも特に、モーションセンサを用いる音楽演奏システムを制作・研究している。

中村滋延 (Shigenobu Nakamura)

1950 年生まれ。作曲家、メディアアーティスト、音楽評論家、映画研究者。交響曲 5 曲を含む 100 曲以上のクラシック系現代音楽を作曲。また「音楽系メディアアート」という領域を創成し、視覚要素を含むコンピュータ音楽やサウンド重視の映像アートを多数制作。現在、九州大学大学院教授（芸術工学府コミュニケーションデザイン科学コース、芸術工学部音響設計学科）。