

振動による採譜

岩手県立一関第一高等学校理数科 3 年情報班

古山颯人 阿部蓮 小保方愛来 菅原七海

要約

楽器でどのような演奏をしたかを情報に起こすことを「採譜」という。採譜には MIDI ドラムなどの専用の機材を使用することが一般的だが、アコースティックドラム本体にセンサを取り付けるだけで簡単にかつ安価に採譜できる仕組み作りを試みた。本取り組みではセンサに Arduino を用いて、その値をコンピュータ上で分析するプログラムを作成し、このような仕組みで採譜が可能であることを確認した。

1 はじめに

採譜とは、人が楽譜の通りに演奏することと反対の操作で、演奏を分析しどのような演奏をしたかをデータに起こすことである。採譜を行うときの規格として世界共通の MIDI がある。MIDI とは、音楽の演奏情報をデータ化し電子楽器やパソコンで再生できるようにしたものである。

従来の MIDI を活用した採譜は、機材が大型で高価なものであったり、楽器を演奏するような感覚でデータの入力できなかつたりするデメリットがあった。

そこで私たちは、センサを用いて実際の楽器を演奏する形で情報を計測し、分析する方法に注目した。楽器を用いるため、計測する情報は音に関する振動とした。振動は振動数、速度、変位、加速度を用いて表され、一度に手に入る情報が多く分析が難しい。実験を単純化させるために、鳴らすことのできる音の種類が少ない楽器を採用することとした。ピアノは88鍵あり、周波数によって音の高さに違いが表れる。また、周波数の違いはあまり

大きくない。そのため、叩くと振動するドラムを実験対象として使うことにした。

河田ら (2019) は、ドラム演奏の音響信号に対する自動採譜手法の精度向上を目的とし、ドラム音源の位相情報を考慮した回帰型ニューラルネットワーク (以下 RNN という) による採譜を行っていた。RNN では文章などの連続的な情報を利用できるが、ドラム楽器の位相情報や打点時刻推定に用いられている。さらに、RNN を用いるために音源の前処理が行われていた。

私たちはリアルタイムでの採譜を目指すためドラムの一つ一つのパーツに振動センサを取りつけ、採譜する方法をとることにした。これによって、実験の対象(楽器の数)が増えた際に叩いた楽器を明確化することで正確性が低下するのを防ぐ目的がある。

2 研究方法

(1) 実験道具

- ・アコースティックドラム (スネアドラム)
- ・パソコン

- ・ 振動センサ (Garosa Garosaihelpmgx4o)
- ・ Arduino
- ・ Arduino IDE

加速度を検出できる接触式の圧電型振動センサを用いた。圧電効果を利用して、振動を検知することができる。

Arduino は設計図や情報が公開されているハードウェアのひとつで、動作に必要な開発システムを Web サイトからダウンロードできるものである。自身の手で変更を加えることもでき、図 1 のようなシステムを製作した。

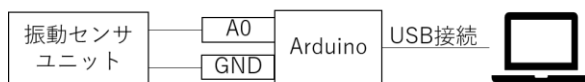


図1: 計測システム接続

計測するドラムは、研究開始時ではスネアドラム、ハイハット、バスドラムの3種類の楽器を用いる計画だった。それらは、ドラム演奏を構成する最も基本的な打楽器で、そのほかのシンバル等は演奏の装飾として用いられることが多い。この理由から、演奏において叩かれることの多い3つを中心に行うことにしたが、複数のドラムを演奏することで他のドラムに影響する可能性を考慮し、ドラムの中で、最も叩く回数が多いとされているスネアドラムのみを用いて実験を行うこととした。

スネアドラムのヘッドと呼ばれる皮の部分にセンサをマスキングテープで固定し、打点の近くで振動を拾えるようにした。

実験に使用した装置を図2に示す。



図2: ドラムと計測装置

(2) 実験・調査の手順

- ・ 一定間隔でドラムをたたく。
- ・ 振動センサでドラムの振動を計測する。
- ・ 計測した情報をモニターに表示する。
- ・ 実際の演奏と表示された結果とを比較検証する。

(3) データ処理の方法

振動センサからの計測信号を「時刻 (t) → 加速度 (m/s²)」に変換するよう図3のプログラムで処理を行う。これによって、楽器を叩いた時刻およびその時の振動の加速度が視覚的にわかるようになる。

```

Dekiru2 §
|
//振動センサを接続するピン
//A0ピンに接続します
const int sPin = A0;

void setup() {
  //2000000bpmでシリアル通信
  Serial.begin(2000000);
}

void loop() {
  //振動センサの値を取得
  int sValue = analogRead(sPin);

  //シリアルモニタに振動センサからの値を表示
  Serial.println(sValue);

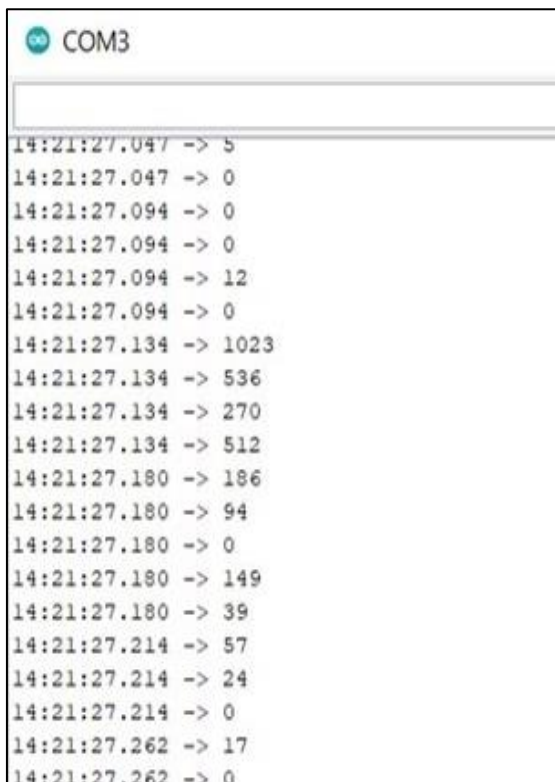
  delay(10);
}

```

図3 : 時刻と加速度を表示するプログラム

3 結果

実験の結果、図4のように、Arduino IDEのシリアルモニタを用いて時刻（t）と振動センサが感知した振動の加速度（ m/s^2 ）を表示することができ、演奏情報をデータ化できることが確認できた。



```
COM3
14:21:27.047 -> 5
14:21:27.047 -> 0
14:21:27.094 -> 0
14:21:27.094 -> 0
14:21:27.094 -> 12
14:21:27.094 -> 0
14:21:27.134 -> 1023
14:21:27.134 -> 536
14:21:27.134 -> 270
14:21:27.134 -> 512
14:21:27.180 -> 186
14:21:27.180 -> 94
14:21:27.180 -> 0
14:21:27.180 -> 149
14:21:27.180 -> 39
14:21:27.214 -> 57
14:21:27.214 -> 24
14:21:27.214 -> 0
14:21:27.262 -> 17
14:21:27.262 -> 0
```

図4:計測結果

この結果には叩いていない時も加速度が $0m/s^2$ として表示されている。図3のプログラムでは、スネアドラムを叩いていない場合でも時刻とセンサの値が表示されていたためである。しかし、本来の目的はどの時刻にどの強さで叩いたかを知ることであり、叩いていないときの情報は不要である。そこで、センサの加速度の値が $150m/s^2$ 未満の場合はスネアドラムを叩いていないとみなし、結果を一切表示しない条件を付けた。これは、計測の

結果、叩いたときのほとんどが $150m/s^2$ を上回ったからである。

また、一回叩いたあとの余韻の振動を10ms後に表示していたため、検出の間隔をさらに50ms（0.05秒）延ばすようにプログラムを改良した。つまり、シリアルモニタに表示される間隔は60ms（0.06秒）となっている。

最終的に図5のようなプログラムで計測を行った。



```
sketch_jul15a $
//振動センサを接続するピン
//A0ピンに接続します
const int sPin = A0;

void setup() {
  //2000000bpmでシリアル通信
  Serial.begin(2000000);
}

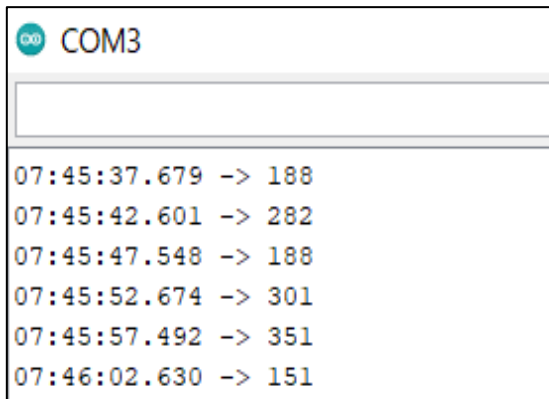
void loop() {
  //振動センサの値を取得
  int sValue = analogRead(sPin);

  //シリアルモニタに振動センサからの値を表示
  if(sValue >150){Serial.println(sValue);
  delay(60);}
}
```

図5:改良したプログラム

改良したプログラムを用いて計測し、図6のような結果が得られた。こちらも時刻と振動センサが感知した振動の加速度を表している。

加速度 $150m/s^2$ 未満のデータを表示していないため、スネアドラムを叩いていないとき（加速度0のとき）が表示されていない。また、叩いた後プログラムを一時中断しているため、余韻の振動が表示されることが少なくなった。



Timestamp	Value
07:45:37.679	188
07:45:42.601	282
07:45:47.548	188
07:45:52.674	301
07:45:57.492	351
07:46:02.630	151

図6:改良後の計測結果

4 考察

実験を行った結果、意図した通りの演奏をリアルモニタに表示できていることが確認できた。また、プログラムに変更を加えて、叩いていないときや余韻の表示を減らすことができた。このことから今後実験の試行回数を増やしていけば、さらに正確性を向上できたり、より複雑な演奏にも対応できたりすると見込まれる。

より正確な結果を得るため、プログラムを中断させる時間や、スネアドラムを叩いていないとみなす加速度の値を変えて反復実験を行ったり、根拠となる文献を探したりする必要がある。

音は周波数で変わることが多いため、加速度から振動を構成するほかの値に変換できると他のことにも活用できると考えた。

センサを取り付ける際に測定物に密着するようにしっかり固定し、フィルタとならないようにするべきであったが、マスキングテープを用いた簡易的な固定方法をとってしまった。しかし、取り外しをしやすいようにするため、固定方法の吟味をする必要がある。

楽譜としての見やすさや採譜のしやすさを重視すると、実験で使用した道具を変更して

実験を行う必要がある。接触型のセンサを使用した、非接触型の振動センサの場合や、加速度以外を検出する振動センサの場合、表示や採譜しやすさに違いがみられるのか調べたい。

5 結論・今後の展望

振動センサを用いてどのような演奏をしたかを視覚的に表示することに成功した。次の段階として、スネアドラムの個数を増やし、スネアドラムでの正確性の向上を目指す。また、今回実験出来なかったハイハットやバスドラムでの実験を行う。その後、センサの数を増やし、同時に演奏する楽器をひとつずつ増やして実験出来るようにする。また、その際、共振により他の楽器のセンサが反応しないように他の楽器の振動を拾うことを防ぐフィルタを作ることで正確性を向上させる。

実験回数を増やしたり正確性を向上させたりするために、プログラムの改善やセンサの固定方法の見直しが必要である。

そして、この採譜方法を発展させ、ドラム以外の楽器でも使用できるようにするためには、音の高さの情報が必要となる。このため、時刻と周波数を表示できるようにしていきたい。

楽器に対するセンサの利用がこの研究の基本だが、リアルモニタに表示される値を自由に変えられるようになれば、楽器以外にも活用できる可能性がある。佐々木ら(2012)によると振動センサを用いた血流状態の測定も高感度で正確に行われている。

謝辞

本研究を行うにあたり、協力していただいた三浦邦広先生、千葉賢一先生、佐々川秀勝先生、ありがとうございました。

参考文献

- ・大塚匡紀 (2014) :MIDI ギターの精度向上を目指した音響信号処理の検討, 情報処理学会研究報告, 6p
- ・星加佑, 及川靖広(2019):スネアドラムの膜と弦の衝突における振動解析, 卒業研究論文ライブラリ, 2p
- ・河田洋人, 保利武志, 中村和幸(2019):位相情報を考慮した RNN によるドラム自動採譜, 情報処理学会研究報告, 4p
- ・佐々木康弘, 高橋尚武, 相本隆司, 源新輝 (2012):微小な振動を検知する超高感度振動センサ技術開発とその応用, Vol.65 No.2 ビッグデータ活用を支える 基盤技術・ソリューション特集, 5p
- ・Metoree (2022) : 【2022年版】振動センサー 5 選/メーカー10社一覧
[https://metoree.com/categories/2051/\(2022.2.15\)](https://metoree.com/categories/2051/(2022.2.15))
- ・ヤマハ「プリント楽譜」:MIDI の基礎知識
[https://www.print-gakufu.com/guide/4005/\(2021.9.15\)](https://www.print-gakufu.com/guide/4005/(2021.9.15))
- ・Arduino :What is Arduino?
<https://www.arduino.cc/en/Guide/Introduction> (2021.6.1)
- dd15038 (2018) :【Arduino】振動を測ってみた, 超ゆるゆる技術ブログ
<https://dd15038.hatenablog.com/entry/2018/07/26/160735> (2021.11.30)