

指揮者の動きに同期した演奏アニメーションの生成

Generation of brass band animation synchronized with the movement of conductor's hand

小林 克樹[†] 村木 祐太[†] 小堀 研一[†]
Katsuki Kobayashi Yuta Muraki Ken-ichi Kobori

1. はじめに

近年, PlayStationVR や HTC Vive, Oculus Rift などの VR デバイスが普及し, 手軽に仮想現実を体験することが可能になっている。

ところで, 音楽データの一つである MIDI データを入力とし, 楽器演奏アニメーションの自動生成を行う研究^[1]が行われている。本研究では, VR デバイスを用いて, 現実のユーザの指揮者の動きと同期した楽器演奏アニメーションの生成を行うシステムを提案する。本システムでは, ユーザの指揮者の腕の動きを取得し, その動きの速さ, および大きさを推定する。推定は取得した速度ベクトルの向きが変化した地点を拍の開始地点とし, その地点を用いることで実現する。そして, 推定したテンポを用いてアニメーションおよび音楽の再生速度を制御する。また, 推定した大きさをを用いて音楽の再生音量を変更する。さらに, 仮想空間内に CG アニメーションの生成を行うことにより, ユーザに対して没入感を与える。

2. 提案システム

本システムは Unity^[2]を用いて実装を行った。本システムのフローチャートを図 1 に示す。

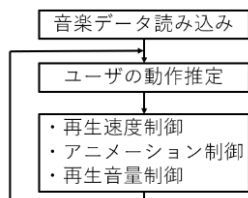


図 1 フローチャート

ユーザは HTC Vive を頭部に装着し, 右手にコントローラを所持し, 指揮者の動作を行う。また, 本システムではコントローラを指揮棒として仮想空間上に表示する。

2.1 音楽データ読み込み

まず, 音楽データの読み込みを行う。本システムでは音楽データとして音楽データの一つである MIDI データを使用する。MIDI データには楽器の種類やテンポなどの情報が含まれており, これらの情報を使用することで演奏アニメーションの多様性を増すことが期待できる。また, MIDI データの読み込みには Unity 内で使用することのできるアセットの一つである「Midi Tool Kit Pro^[3]」を用いた。このアセットは MIDI データを音源として再生することができる点や, 前述した MIDI データに含まれる情報を解析することが可能である。

そして, MIDI データから取得した楽器の種類の情報から演奏者の表示を行う。なお, 対応している楽器はフルート, ギター, クラリネット, トランペット, ドラム, ピアノの 6 種類である。

2.2 ユーザの指揮動作推定

次に, ユーザの指揮者の動作からその動作の速さと大きさを推定する。今回は最も曲数が多い 4 拍子の曲に注目し, ユーザの動作を定義した。ユーザの動作を図 2 に示す。

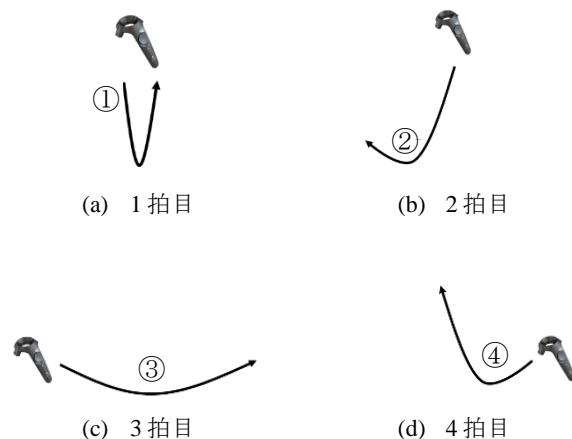


図 2 ユーザの動作

2.2.1 動作の速さ推定

図 2 に示すそれぞれの動作にかかった時間を一拍の実時間として取得する。ここで, 一拍の実時間とテンポの関係を式(1)に示す。

$$\text{一拍の実時間(s)} = 60(\text{s}) / \text{テンポ} \quad (1)$$

同式より, 以下の式(2)を得ることができる。

$$\text{テンポ} = 60(\text{s}) / \text{一拍の実時間(s)} \quad (2)$$

式(2)より, 一拍の実時間を取得することでテンポの推定を行うことが可能である。

一拍の実時間を取得するためには, 図 2 に示したそれぞれの動作の開始地点を取得する必要がある。開始地点の取得には位置情報から算出することのできる速度ベクトルを用いる。速度ベクトルの向きが右から左, 左から右, 上から下に変化した地点をそれぞれの動作の開始地点として取得する。それぞれの地点間の動作時間を一拍の実時間とし, 式(2)を用いることでテンポの推定を行う。そして, 推定したテンポを MIDI データから取得したテンポで除算することによって再生速度倍率を決定する。

2.2.2 動作の大きさ推定

次に、ユーザの指揮者の動きからその動作の大きさを推定する。動作の大きさには前節で求めた開始地点とそれぞれの動作の経過地点を用いて算出する。経過地点は速度ベクトルの向きが下から上に変化した地点を取得する。そして、動作の開始地点から経過地点までの距離と経過地点から次の動作の開始地点までの距離の合計値をその動作の大きさとして算出する。

2.3 再生速度倍率の調整

動作の開始地点で音楽やアニメーションの変更を行っているため、算出した再生速度倍率をそのまま使用すると、MIDI データと指揮動作の拍の位置にずれが発生する。そのため、再生速度倍率の調整を行う。

例として、前の動作時間が t 秒、テンポ tmp 、再生速度倍率 v と推定されたとする。この t 秒まではテンポ tmp で再生を行う。そして、 t 秒を超えた後はその時点で経過地点を取得しているかどうかで再生速度倍率を決定する。経過地点を取得している場合は式(3)を用いて加速度的に変化させる。取得していない場合は式(4)を用いて等速に変化させ、取得した後は式(3)を用いて変化させる。

$$\begin{aligned} \text{再生速度倍率} &= 1 / (ax + b) \\ (a &= (1 - 10v) / (t - t_{min})v) \\ (b &= 10 - t_{min} * a) \\ (t_{min} &= 60 / (0.1 \times def)) \end{aligned} \quad (3)$$

$$\text{再生速度倍率} = \text{再生速度倍率} - 1.2 / def \quad (4)$$

式(3)の x は動作時間、 t_{min} は再生速度倍率が最小値になる動作時間、 def は MIDI データから取得したテンポを示している。なお、再生速度倍率の最低値は 0.1 である。例として、 $t = 120$ の時の再生速度倍率の変化を図3にグラフで示す。

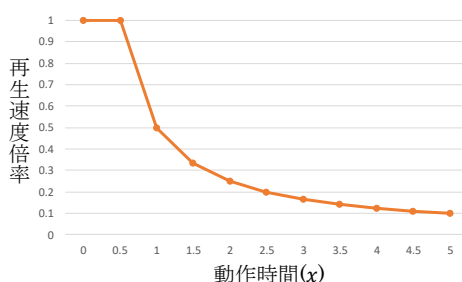


図3 再生速度倍率変化の例

2.4 再生速度、再生音量、アニメーションの制御

最後に、推定したテンポから得られた再生速度倍率および動作の大きさを用いて音楽再生速度、再生音量、アニメーションの制御を行う。動作の大きさは MIDI データの再生が始まる前に、ユーザに指揮動作を 5 回行い、それぞれの動作の平均の大きさを求め、動作の大きさの基準とする。そして、推定された大きさを基準とした大きさを除算することによって再生音量を決定する。アニメーションは

HTC Vive^[4]を用いて仮想空間上に投影される。仮想空間上に投影されている様子を図4に示す。



図4 本システムのイメージ

同図右上の矢印部分に存在する白い棒は HTC Vive のコントローラに追随する指揮棒である。これらの処理を MIDI データの再生が終了するまで繰り返す。

3. 実験と考察

実験として、本システムを 20 代の男女含む 12 人に使ってもらい、動作の速さおよび大きさによって音楽やアニメーションが変化しているかどうかの確認を行った。

実験結果として、被験者の意見から正しく音楽やアニメーションの変更を行うことができていることが確認できた。これにより、ユーザの指揮者の動作から速さや大きさが推定され、それが音楽やアニメーションに正しく反映できていると考えられる。

しかし、得られた意見として、「遅くした際に音楽と動作にズレを感じる」や「本来の曲のテンポがわからない」という問題が明らかになった。前者の意見に関しては、図3より再生速度倍率を急激に下げている部分がある。そこから、拍の位置が取得された後に算出される再生速度倍率が急に上がったことがズレを感じてしまう原因であると考えられる。そのため、再生速度倍率の変化を緩やかにすることで改善することを考えている。後者の意見に関しては、動作の大きさの基準を取得している際に、MIDI データから取得したテンポに合ったメトロノームの音を流すことにより改善できると考えられる。

4. おわりに

本研究では VR デバイスを用いて、ユーザの指揮者の動作からその速さと大きさを推定した。そして、推定した速さを用いて音楽再生速度およびアニメーションを制御した。さらに、推定した大きさを用いて音楽の再生音量を変更するシステムの開発を行った。今後の課題として、実験から得られた再生速度倍率を変化する式の改善が挙げられる。また、今後の展望として 4 拍子以外の曲に対応することが考えられる。

参考文献

- [1] 武内航, 堀井絵里, 藤代一成. "吹奏アニメーションにおける音源同期型姿勢制御." 情報処理学会第 80 回全国大会講演論文集 2018.1 (2018): pp.315-316.
- [2] Unity <<https://unity.com/ja>>, <最終アクセス 2020/01/09>.
- [3] Midi Tool Kit Pro - Asset Store <<https://assetstore.unity.com/packages/tools/audio/midi-tool-kit-pro-115331>>, <最終アクセス 2020/01/09>.
- [4] VIVE™ <<https://www.vive.com/jp/>>, <最終アクセス 2020/01/09>.