

ハンドベル自動演奏のための 3 次元操作インタフェースの開発

奥田佳史 *

電子情報技術科総合制作テーマとして「ハンドベル自動演奏のための 3 次元操作インタフェースの開発」の立案及び指導を行った。センサーとして非接触型 3 次元ポインティングデバイスを採用し、指の位置情報からリアルタイムにハンドベル自動演奏装置を制御するマンマシンインタフェースの構築を目指した。マルチコア機能とマルチタスク同期機能により並列処理を実現し、指揮に倣い演奏中の曲のテンポの揺らぎにより演奏能力を向上させた。演奏能力は近畿ポリテクビジョン 2016 作品展示部門において、来場者投票による優秀作品賞受賞として認められた。
Keywords : 非接触型 3 次元ポインティングデバイス, 非同期 I/O, タスク間同期, マルチコア。

1. 緒言

ハンドベル自動演奏装置は平成 26 年度の総合製作課題[1]であり、midi 形式の楽譜データを解析し、合計 27 個のハンドベルで演奏する装置である。演奏時には midi データに含まれるテンポ情報をそのまま利用していたために演奏のテンポは毎回固定されたままであり、微妙なテンポの変化による楽想を表現することができなかった。そこで、ハンドベル自動演奏装置にテンポ等を指示するマンマシンインタフェースを装備し、操作者と対話的に演奏を変化させる機能を追加し、より表現力ある演奏装置の開発を目指した。

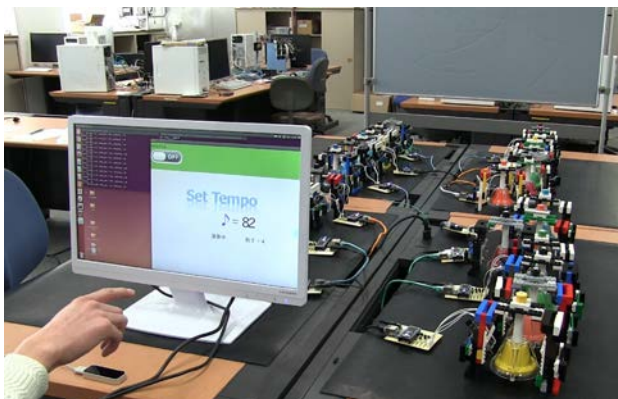


図 1 指先で操作中のハンドベル自動演奏装置

指揮者のように指先の合図により、演奏の開始・終了及びテンポの揺らぎ等の音楽のもつニュアンスを再生できる装置の開発を目指した。以後、システムの操作者をここでは指揮者と呼ぶことにする。動作指示やテンポを調整するためのインタフェースとして革新的な 3 次元非接触型ポインティングデバイスである Leap Motion コントローラ[2]を採用した。このインタフェースにより、指揮者は楽曲のテンポを自由にコントロールすることができる。図 1 に指先で演奏中のシステム全体の外観を示す。

2. システム概要

2.1 システム制御フロー 図 2 にシステムの制御フロ

ーを示す。指揮者の指の動きを Leap Motion コントローラは指の位置座標をクライアントに最大 100FPS の間隔で送信する。クライアントはテンポ情報に変換し、midi データに基づいたタイミングでハンドベルを叩く機能をもつソレノイド制御サーバにイベントを送信する。ソレノイドはイベント発生に従い遅延なくソレノイドを叩き発音を完了する。

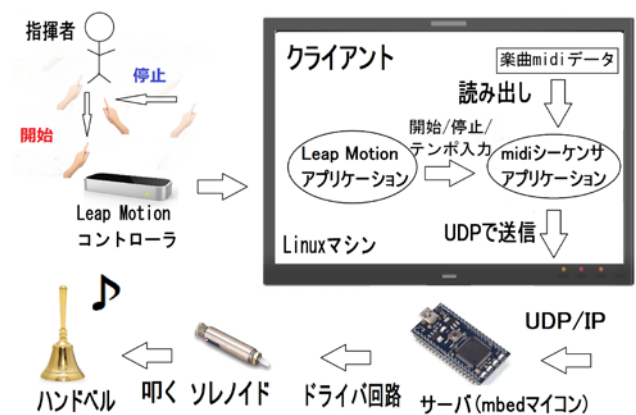


図 2 システム制御フロー

2.2 非接触型 3 次元ポインティングデバイス Leap Motion コントローラは 2012 年に Leap Motion 社から販売された手のジェスチャーによりコンピュータの操作を行うことのできるデバイスである。Leap Motion コントローラの実物は手のひらにのるほどの大きさとなる。2 基の赤外線カメラと赤外線照射 LED から構成されており、赤外線 LED に照らされた手や指を 2 基の赤外線カメラで撮影し、画像解析によって対象物の位置の視差を利用して対象物までの距離を計算。これにより指の位置を 1/100 mm 単位で検出し、3D 空間での手や指の位置を割り出す。検出できる範囲は半径 50 センチ程度、中心角 110 度の空間で、手、指、ペンのようなポイントを指し示すツールを認識する。

2.3 Interaction Box Leap Motion コントローラはその中心から、垂直方向に Y 軸、水平方向に X 軸、奥行き方向に Z 軸の検出範囲を持つが、この中に Interaction Box と呼ばれる直方体の仮想検出環境を持

* 電子情報技術科

ることができる．図 3 に Interaction Box の座標系を示す．

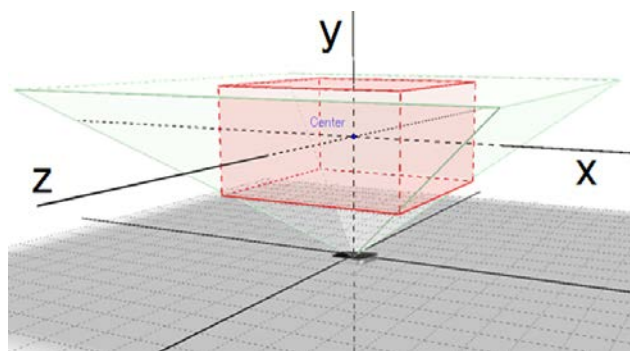


図 3 Interaction Box の座標系

Interaction Box クラスの `normalizePoint()` というメソッドを使えば，Interaction Box 内に指，もしくは手を認識させた際にそれらが Interaction Box 内のどこにあるのかを Interaction Box の左下奥を原点として三次元それぞれの座標として 0 から 1 の数値で取得することができる．

この三次元の直方体の X 軸方向と Y 軸方向のみを利用することで平面座標として扱うことができ，この値をウィンドウのサイズに掛け合わせることで指や手がウィンドウのどこに存在するかを知ることができる．

3. 仕様

本システムのマン=マシンインタフェースで実装する動作仕様を表 1 に示す．

表 1 動作仕様

意図	ユーザの動作
曲開始・再開	指先で拍のテンポを入力する
曲一時停止	Leap Motion コントローラの上に手をパーにして持って行き，手のひらを下にしてグーにすることで曲を一時停止させることができる
曲選択ボタンを押す／曲ディレクトリを移動する	人差し指を選択ボタンの上に持っていく，Leap Motion コントローラより奥に押し込む（以下，この動作をタッチと呼ぶ）

4. 開発環境・動作環境

推奨環境及び開発の容易性を考慮し本システムでは以下の開発環境・動作環境を採用した．

CPU : i5-4460 3.2GHz 4Core

OS : Ubuntu 14.04 LTS

SDK : Leap Motion SDK 2.3.1

開発言語 : C++

画像ライブラリ :

SDL(Simple Direct Media Layer) 2.0 [3]

5. モジュール構成

図 4 に本システムのモジュール構成を示す．

5.1 入力モジュール 入力モジュールには Leap Motion, 音源モジュールにはハンドベル演奏装置を使用している．

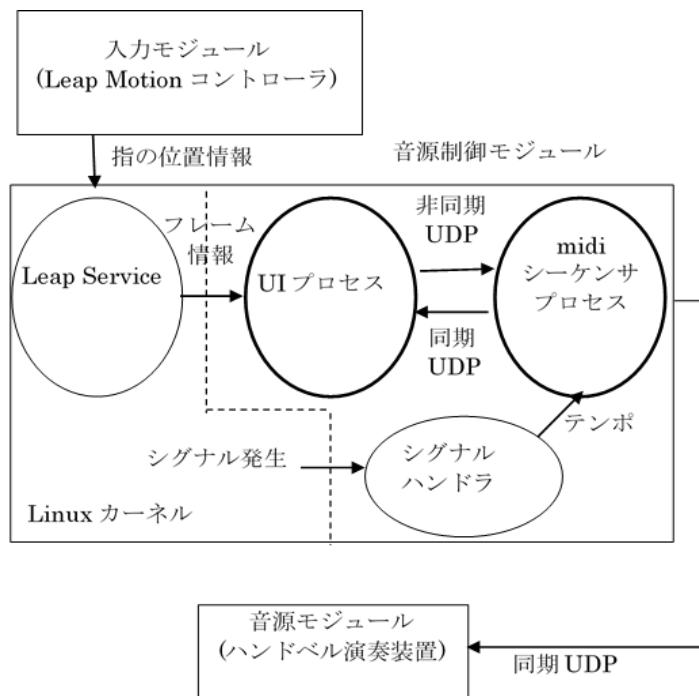


図 4 モジュール構成図

5.2 音源制御モジュール 音源制御モジュールは UI プロセスと，midi シーケンサプロセスから構成されている．UI プロセスでは Leap Motion からの指の位置情報を入力させるためのユーザインタフェースを提供し，指の位置情報から開始，停止，テンポ変更等の動作の判別を行う．midi シーケンサプロセスでは midi データの解析を行い，音源モジュールと通信を行って 27 個のハンドベルのうちどのハンドベルを鳴らさせるかという指示を出す．UI プロセスと midi シーケンサプロセスは UDP による同期を取りながら動き，UI プロセスから midi シーケンサプロセスへの動作指示はシグナルハンドラによる割り込み処理を行わせることにより実現している．

5.3 音源モジュール 昨年度の総合製作課題のサーバ機能をそのまま利用した．

6. 音源制御モジュール

6.1 タスク構成 (1) UI プロセス SDL により，曲選択画面と曲制御画面を提供する．Leap Service とやり取りをし，指の位置やテンポの取得，動作指示を行う．指のポインタは「Interaction Box」の検出環境内に右手人差し指が存在するとき，InteractionBox クラスの `normalizePoint()` メソッドで Interaction Box 内の座標を求め，その数値に応じて描画を行う．ボタンや曲名のカードを押したいときは，ボタン，カード上に指のポインタをあわせ，押し込むことで選択できる．これは，

Leap Motion コントローラより指が奥にあるとき、Pointable クラスの touchDistance()メソッドにより、タッチ状態であると判別されるためである。またテンポは、過去 10 回のテンポの単純移動平均を取ることで、指揮のテンポを滑らかに変化させることを実現している。

(2) midi シーケンサプロセス midi 形式のデータを先頭から順番に解析するプログラムである。UI 部から送られてきたテンポ情報をもとに、音源モジュールへの通信を制御する。

6.2 非同期 I/O プロセス間通信 UI プロセスと midi シーケンサプロセスはソケットを用いた非同期 I/O プロセス間通信を行っている。SDL は描画のループを作る必要があり、midi シーケンサでは sleep 関数を使用しているので、シングルタスク構成の場合、sleep しているとその間 CPU が動くことができず、描画ができない問題が生じるからである。プロセス間通信のシーケンス図を図 5 に示す。

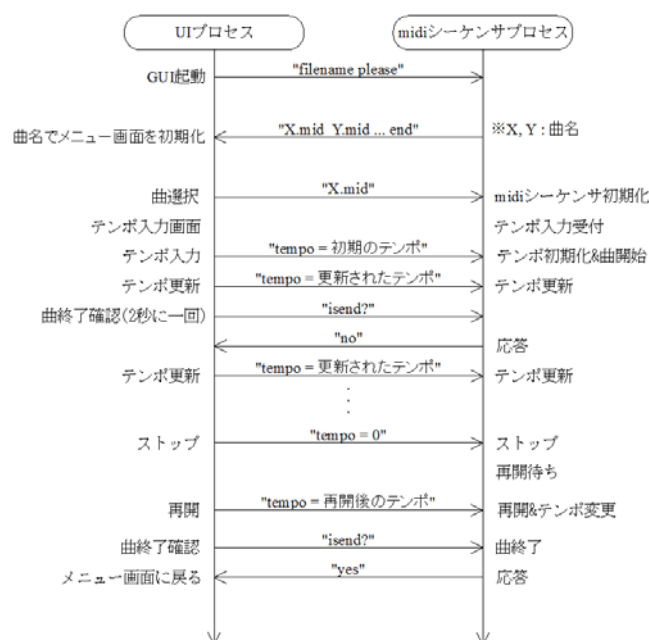


図 5 プロセス間通信シーケンス

またマルチコアプロセッサでない場合は、例えばマルチタスクにしても OS は 1 CPU に時分割で資源を割当てただけで、SDL の負荷に応じて midi プロセスの指定した sleep 時間は保障されなくなる。つまり描画のために曲のテンポが間延びする現象が発生し、初心者のように演奏時に楽譜追いかけるだけのようなどたどしい演奏になってしまう。よって本システムでは各プロセスを独立した CPU コアに割当てて必要がありマルチコアプロセッサは必須の要件となる。sleep 関数と描画を両立させるために、midi シーケンサプロセスと UI プロセスを独立した 2 プロセスで構成し、UDP 通信することでプロセス間通信を行った。midi シーケンサプロセスは非同期通信を行い、データ受信があればシグナルハンドラを実行する。受信がなければ midi データの解析、sleep を続行する。

7. 操作系

指揮者が操作を行う曲目選択画面の構成を図 6 に示す。

7.1 曲目選択 曲目選択画面で Leap Motion コントローラに右手を認識させると青色のポインタが表示される。ポインタは Leap Motion コントローラの真上より奥に押し込むと赤色のポインタに変わり、この動作をタッチと呼ぶ。

水色のカードのアイコンは曲のカードを表し、このカードの上にポインタを持って行き、タッチすることで曲が選択でき、曲制御画面に移行する。黄色いアイコンは曲のディレクトリで、このアイコンにタッチするとディレクトリを階層的にする。次の曲目、前の曲目選択ボタンをタッチすると、次の 5 曲、前の 5 曲が表示される。



図 6 曲目選択画面

7.2 演奏中の状態遷移 曲目選択から始まるシステム状態遷移図を図 7 に示す。状態遷移のイベントを①～⑦に示す。

- ① 曲名カード：選択するとその曲の演奏制御画面へ
- ② 指のポインタ：指の位置を示すポインタ
- ③ 前の曲目選択ボタン：前の 5 曲表示
- ④ 次の曲目選択ボタン：次の 5 曲表示
- ⑤ 曲名：現在演奏中の曲名
- ⑥ 表示切替えボタン：二拍子と最適拍子を切り替える
- ⑦ menu ボタン：演奏を中断し menu に戻る

7.3 テンポ操作 図 7 にテンポ操作画面を示す。曲制御画面では曲の開始、一時停止、テンポ変更ができる。まず青いポインタを出現させる。menu に戻るボタンをタッチすると曲選択画面に戻る。

拍子切り替えボタンを押すと、四拍子、三拍子入力画面から二拍子入力画面に切り替わる。velocity0 の on/off ボタンは、タッチすると ON/OFF を切り替える。このボタンは midi データに含まれる velocity = 0 のノートオンイベントを鳴らすかどうかの切り替えである。鳴らすなら ON、鳴らさないなら OFF になっている。

黄色い星は待機中と一時停止中、演奏中で指を認識していない時に表示され、待機中と一時停止中は曲の元のテンポを表す速さで動き、演奏中は入力されたテンポを表す速さで動く。テンポの入力方法は Tap 領域に指を入れることで Tap 領域が移動する。追いかける速さによって曲のテンポが入力される。テンポが入力

されると、曲が開始される。曲選択画面の状態遷移を図 8 に示す



図 7 テンポ入力画面

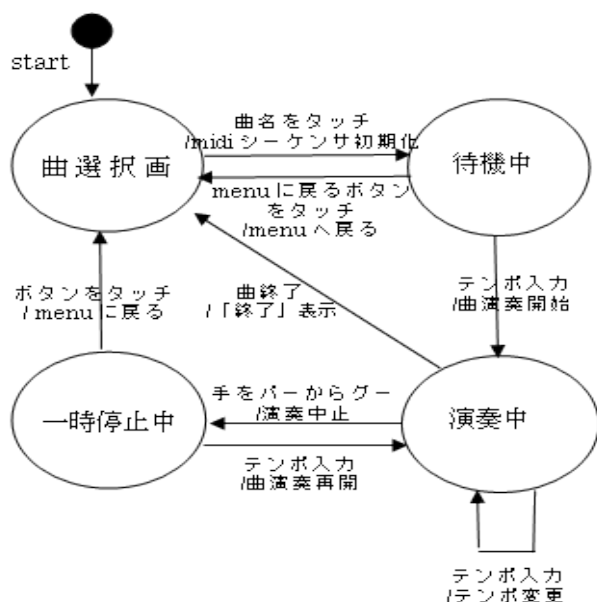


図 8 システム状態遷移図

7.3.1 四拍子の場合 (1)の状態が何も入力されていない状態で、指で Tap 領域を叩きかけること(2)→(3)→(4)→(5)→(2)→(3)→(4)→(5)…と状態遷移していく。この動作を繰り返すことで曲のテンポが入力できる。また、途中でテンポ入力をやめると 最後に入力されたテンポで演奏が続く。テンポ入力をやめるには、Leap Motion コントローラの認識範囲から指をはずせばよい。四拍子入力画面の遷移を図 9 に示す。

7.3.2 二拍子の場合 二拍子入力モードではタップ領域に指を出し入れすることで状態が遷移する。最初は(1)から始まり(2)→(3)→(4)→(5)→(4)→(5)…と状態遷移し、(4)の状態になるとテンポが入力される。二拍子入力画面でも、途中でテンポ入力をやめた場合、最後に入力されたテンポを保って曲が演奏される。二拍子入力画面の遷移を図 10 に示す。

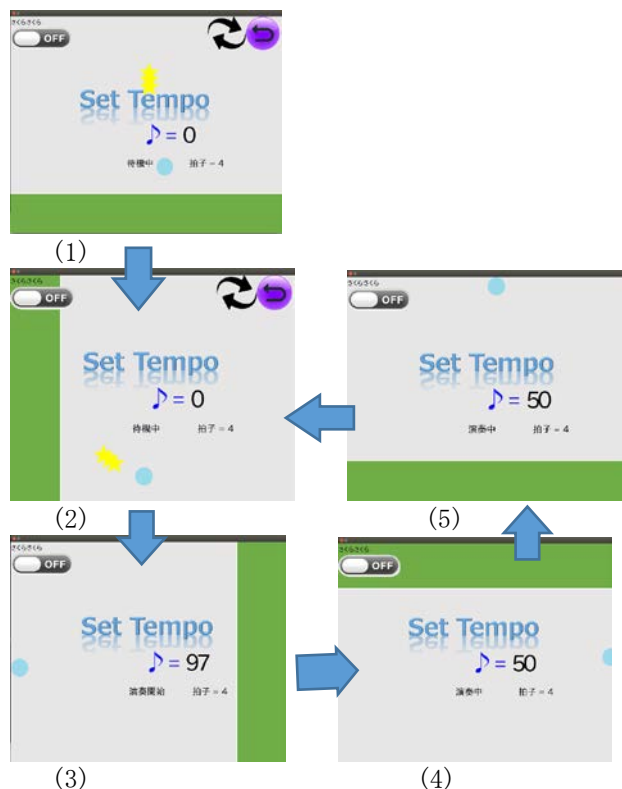


図 9 四拍子入力画面の遷移

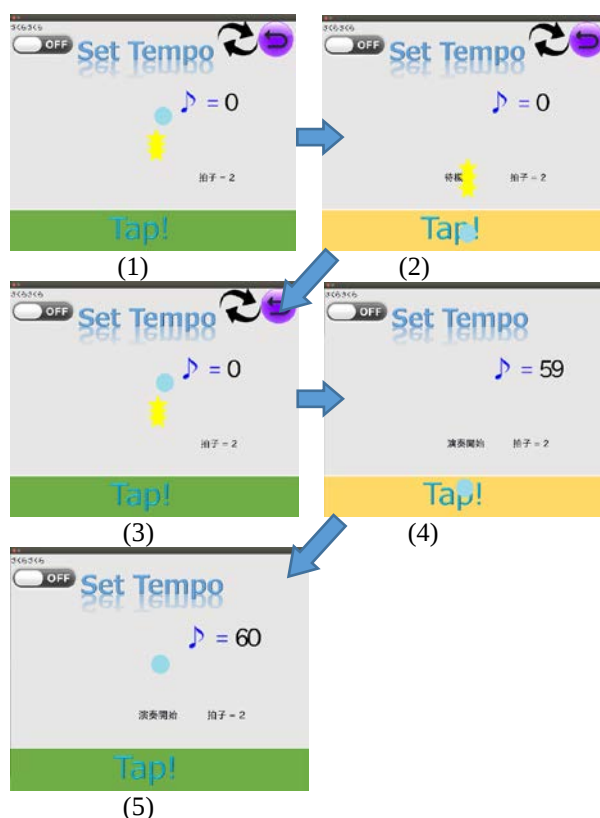


図 10 二拍子入力画面の遷移

7.4 テンポ計算方法 テンポは過去 10 回のタップ時間差の単純移動平均を取っている。移動平均を求めることによって、bpm の敏感な変化により、テンポ変化がダイレクトに伝わってしまい曲が不安定に聞こえるのを防ぐためである。試行結果により、回数 10 が一番安定して曲のテンポ変化を自然に伝えているという

主観に基づいて決定した．図 11 にテンポ取得のための時間測定ポイント及び計算式を示す．

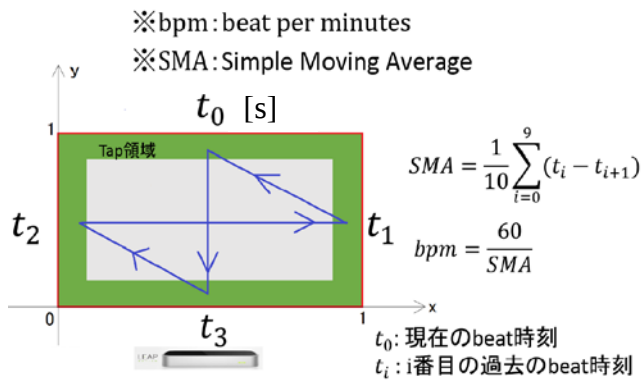


図 11 テンポの取得

8. 動作検証

UI プロセスをフル稼働させ，1000 ms ごとに音を送り，パケットの送信間隔を測定した結果，送信間隔は平均 1000.148 ms であり，人間が遅れを感知できる限界であるとされる 20 ms と比較して，無視できる誤差であることが確認できた．また，曲の拍と指先のタップのタイミングが完全に一致していることを確認した．

人と対話的に演奏をリアルタイムに制御できるシステム

9. 結言

ヒューマンインタフェースを接続することによって、ムを総合課題として立案した．マルチタスク，非同期 I/O に関する知識，C++によるライブラリ操作等の学習項目について開発を通して深く理解できる課題となった．本テーマを熱心に取り組み担当してくれた現生産情報システム技術科 1 年生の塚本隆三君と尾崎真喜一君に感謝します．また本課題の成果として近畿ポリテクビジョン 2016 作品展示部門において、来場者投票による優秀展示作品賞の受賞として認められた．

10. 今後の展開

今後は、楽曲演奏の 2 大要素であるテンポとダイナミクスのうちテンポの変化機能は完成した．もう一方のダイナミクスは演奏中の音の強弱をリアルタイムに変化させ実現していく予定である．

文献

- [1] 近畿能開大ジャーナル，第 23 巻，“ハンドベル自動演奏装置の製作”，pp.51-54，2015．
- [2] <https://developer.leapmotion.com/>
- [3] <https://www.libsdl.org/>

(2016 年 06 月 07 日)