

ジェスチャーによる自動車ロボットの制御

2015SC073 大池綾香

指導教員：大石泰章

1 はじめに

高齢者や身体障害者にとってわずかな歩行も困難になることがある。そうしたとき自分は動かずにロボットに指示することが可能であれば、便利であると思われる。指示する方法としては声やジェスチャーなど様々あるが、声では「上」「少し上」「もっと上」などと言わなくてはならない細かい指示もジェスチャーを使えば簡単に行うことができる。

本研究ではジェスチャーにより指示を行ってロボットを動かすことを試みる。具体的には赤外線により手の動きを読み取る入力デバイス Leap Motion を用いて、読み取った結果を PC 経由でマイコンボード Arduino に送る。これにより、Arduino を搭載した小型の自動車ロボット Zumo を手の位置や指の本数に基づいて動かすことを考える。

2 作成するシステム

2.1 システムの構成

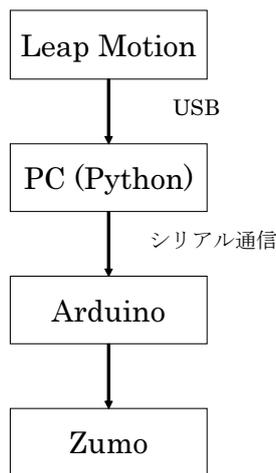


図1 システムの構成

本研究で作成するシステムの構成は図1のとおりである。手の位置や指の本数を Leap Motion で読み取り、その値を PC 上でプログラミング言語 Python を使って処理する。処理した結果をシリアル通信で Zumo に搭載された Arduino に送り、Arduino から指示を出して Zumo を動かす。

2.2 Leap Motion

Leap Motion とは「手」や「指」の検出に特化したセンサである。小型の USB 装置であり、2基の赤外線カメラと3つの赤外線 LED から構成されている。これにより赤外線 LED に照らされた手や指を赤外線カメラで撮影し、

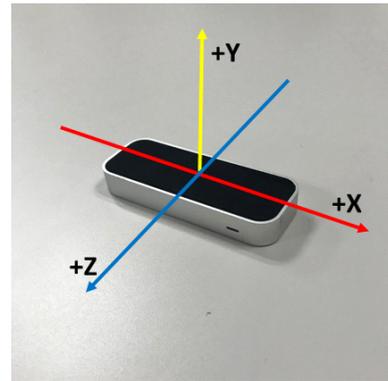


図2 Leap Motion

画像解析によって手のひらの重心の位置を図2の XYZ 座標系に関して測定することができる [1]。ただしこの座標系の原点はリープモーションコントローラの中央に配置されている。 X 軸と Z 軸は水平面内にあり、 X 軸は装置の長手方向と平行、 Z 軸はこれに垂直でユーザの方向を向いている。 Y 軸は鉛直上向きである [2]。

2.3 Arduino と Zumo

本研究では Arduino を搭載した Zumo を使用する。Arduino とは AVR マイコン、入出力ポートを備えた基盤で、PC 上で C 言語に似たプログラミングが可能である [3]。PC でコンパイルした後、USB ケーブルを介したシリアル通信でプログラムを転送することが可能である [4]。

この Arduino を使い、小型の自動車ロボット Zumo の左右のモータを動かすことで、Zumo を動作させる。

3 ジェスチャーとその意味

この章では Leap Motion で読み取るジェスチャーとその意味を述べ、次章で実験結果を述べる。本研究では手の XZ 座標と指の本数を組み合わせてジェスチャーとする。

指の本数が0の場合、すなわちじゃんけんのグーの状態のとき、手がどの場所にあっても Zumo は停止させる。

指の本数が2の場合、すなわちチョキの状態のとき、手の位置の X 座標が負のときは Zumo をその場で左周りさせ、正のときはその場で右周りさせる。

図3は指の本数が5の場合の指示である。左上に手の位置の Z 座標と、右および左のモータに送る値との対応関係を示す。すなわち右の図の青色系の範囲で手の位置を動かすとき、 Z 座標が負ならば Zumo を前進させ、正ならば後退させる。 Z 座標の絶対値が大きいほど、スピードが速くなるようにモータに値を送る。左下に手の位置の X 座標と、右および左のモータに送る値との対応関係を示す。すなわち右の図のオレンジ系の範囲で手の位置を動かすと

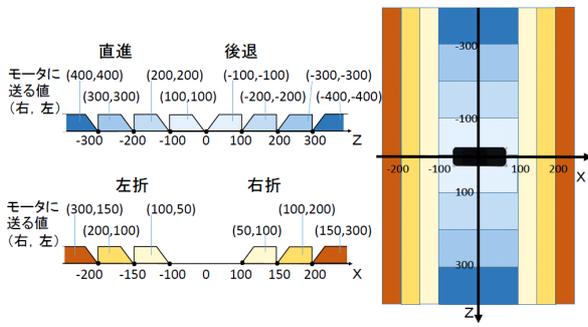


図3 指が5本の場合

き， X 座標が -100 mm 以下ならば Zumo を左折させ， X 座標が 100 mm 以上ならば右折させる．このときも， X 座標の絶対値が大きくなるほどスピードが速くなるようにモータに値を送る．

4 実験結果

2章のシステムを使い，3章で述べたジェスチャーによって，Zumo を制御する実験を行った．

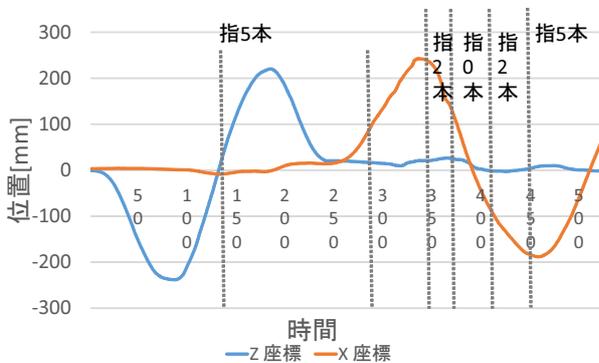


図4 手の位置の座標

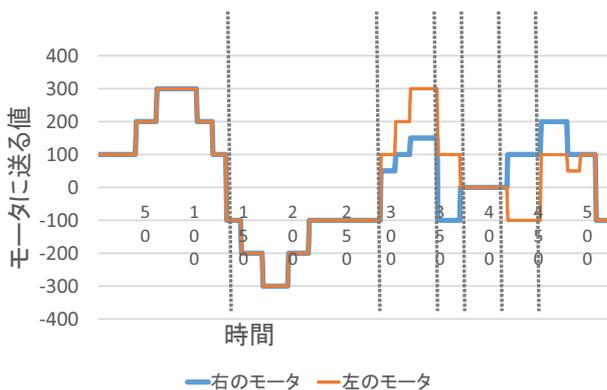


図5 モータに送る値

図4のグラフは，横軸がサンプル数で測った時間を，縦軸が手の位置の座標を表している．青い線は Z 座標の値を示し，赤い線は X 座標の値を示している．

図5のグラフは，横軸が図4と同じくサンプル数で表した時間を，縦軸が Zumo の左右のモータに送る値を表して

いる．青い線が右のモータに送る値を示し，赤い線が左のモータに送る値を示す．

時刻0から350では指の数を5本としている．はじめに，時刻0から130のときは Z 座標が負なので直進しており，座標の大小に応じて速度が増減している．次に時刻130から280のとき， Z 座標が正なので後退しており，やはり座標の大小に応じて速度が増減している．時刻280から350のとき， X 座標が正なので右折しており， X 座標が大きくなるにつれて速度が大きくなっている．時刻350から370の範囲では，指は2本としており，手の X 座標が正なのでその場で右周りをしている．時刻370から420の範囲は指の数を0本にしている．この範囲では図4に見るように手の位置は変化しているが，図5に見るように左右のモータには値0が送られている．時刻420から450の範囲では，指は2本としており，手の X 座標が負であるので左周りをしている．時刻450以降では指は5本としており，手の X 座標が負であるので左折をしている． X 座標が大きくなるにつれて速度が小さくなっていることにも注意する．

以上より，前章で示したジェスチャーで Zumo を制御することが可能ということがわかる．

5 おわりに

手の位置と指の本数を Leap Motion で読み取り，それに基づいて Zumo を動かすことに成功した．

今回シリアル通信で Arduino に値を送るとき有線で送信していたので，Zumo を動かせる範囲がコードの長さによって制限されてしまった．今後はもっと自由に動かすことができるように，無線で通信できるとよいと思われる．また，今回は Leap Motion で読み取った値すべてを使って制御していたが，たまに外れた値を読み取ることがあったので，一定の時間範囲で読み取った値の最頻値をとったり，平均値をとったりといった工夫も必要だと感じた．

参考文献

- [1] 中村薫：『LeapMotion プログラミングガイド』．工学社，東京，2016．
- [2] Leap Motion
https://developerarchive.leapmotion.com/documentation/java/devguide/Leap_Overview.html
- [3] 鈴木美朗志：『Arduino でロボット工作をたのしもう！』．秀和システム，東京，2014．
- [4] 柴田淳：『みんなの Python』．SB クリエイティブ，東京，2018．