

手のジェスチャーを認識して駆動する外部装置の構築

Composition of Remote Control System driven by Hand-Gestures

1205069 豊島 翔 (プロセッサ回路の設計・制御研究室)

(指導教員 綿森 道夫 准教授)

1. 本研究の目的

言葉の壁がある場合でも手の動き、体の動き(身振り手振り)は通じるといったことがあるように、動きでこちらの意図を理解させたい。例えば手話における手の動きを認識して翻訳したいということもあるだろう。翻訳機のようなモノまではいかなくても、定めた手の動きを認識して動作するプログラム、装置を開発することができれば、そういったモノへの自分の中での足掛かりの様なモノになるのではないか。本研究では上で述べたような、手や指の骨格を検出できる Leap Motion を用いて指や手の動きをプログラムで判断し、いくつかの簡単な手話を検出して表示[1]させたり、離れた場所の端末にデータを送り、駆動するシステムの構築を目的とした。

2. TWE-Lite と PIC による端末の構成

シリアル通信やモーターの制御もできて大きさも手ごろで扱いやすい PIC16F688 を用いて TWE-Lite DIP とやりとりをすることにした。PIC とシリアル通信するために TWE-Lite DIP と MoNoStick 側の設定を変更した。MoNoStick から送られてきた情報を TWE-Lite DIP から受け取るために UART 送受信のピンを使用する。PIC の TX ピンを TWE-Lite DIP の RX ピンに、TWE-Lite DIP の TX ピンを PIC の RX ピンに接続する。TWE-Lite DIP 側は後は電源の土を接続して、20 番ピンをグランドに落とすだけである。PIC に対しては RC0 と RC1 をモーターの制御に使用し、12~14 番ピンと 4 番ピンはプログラムの書き込みで使用した。RC2 と RC3 は LED の点灯に割り振っている。

TWE-Lite DIP を TWE-Lite-R を用いて PC に接続すると COM ポートとして認識されるので、そこで TWE-Lite プログラムを用いてシリアル通信専用アプリを TWE-Lite DIP と MoNoStick に書き込む。その状態で TeraTerm などの COM ポートの状態を確認できるものを立ち上げる。「+++」と打ち込みインタラクティブモードに入り、入力された情報がそのまま転送される透過モードに設定する。[2]本研究では透過モードを使用する。これは PIC 側で RS232C(UART)の内臓ハードウェアを使用するのに適しているからである。図 1 に実際の配線の様子を示す。Leap Motion の指の動きに応じて、モーターの正回転と逆回転、LED の点灯消灯が制御可能となった。

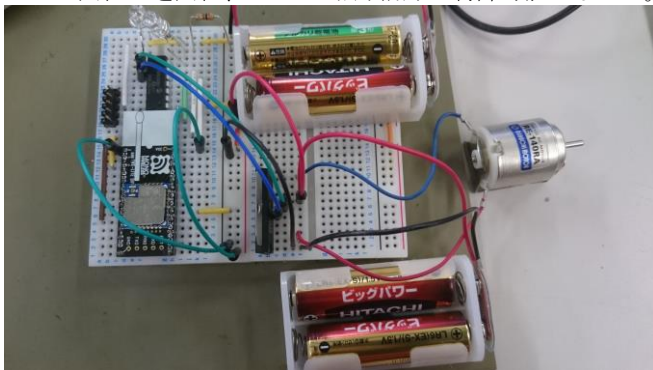


図 1 配線の様子

3. 最終作品

最終作品では、手話のうち”どういたしまして”と”こんばんわ”について Leap Motion で認識してディスプレイ上に表示することにした。Leap Motion の SDK で与えられるそれぞれの関数を試行錯誤しながら 1 つ 1 つプログラミングしていくことで、手や指の動きについてのプログラミング方法が判明し

てきた。更に Leap Motion でどんな動きを検出したときにシリアル通信でデータを送るかということについても検討した。手話の前進と、後進がそれぞれ「前へ手を突き出す」と「後ろへ手を引く」だったのでそれをそのまま使用することにした。つまり手を前に出せばモーターが前へ回転し、後ろへ引けばモーターが逆回転する。そして手を左右に振るとモーターが止まるようにした。手指が認識されていないときは画面の文字を白文字で表示し、手指が認識されると文字に色が付くようにして、認識の有無を分かりやすくした。またどの手指の動きも認識していないときは「No sign」と表示しているが、各手話や動きを認識したらそれに対応する部分が「Detect!」と表示されるようにした。認識されていないときの様子を図 2 に、認識されたときの様子を図 3 に示す。図 3 で表示されている赤い線はスワイプジェスチャーの軌跡であり、左右の動きを認識したので「どういたしまして」のところに「Detect!」と表示されている。

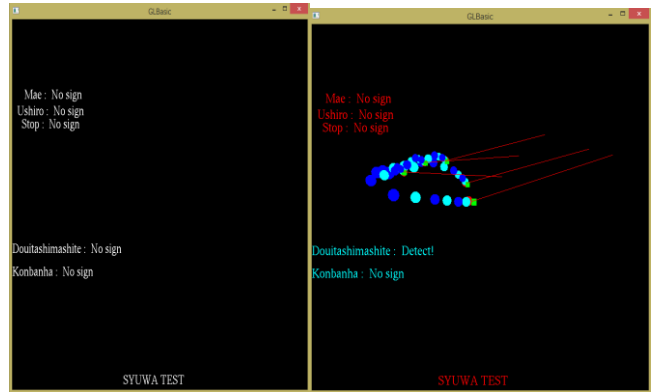


図 2 認識していない状態

図 3 認識している状態

4. 考察

Leap Motion を自作の装置などと組み合わせて使用することにより、利便性や、できることがグンと広がり、さらに自由なプログラムや装置の構築ができるということが分かる。今回はスワイプジェスチャーを最終的に使用していったが、他にも認識可能なジェスチャーはあるのでそれらと組み合わせることによってさらに複雑な手の動きなども正確に検出できるのではないかと考える。全体的な作品の開発を通して左手、右手の検出及びどちらかの手のみを表示することができるようになった。指に関しても全ての指の検出と識別に成功した。指関節の検出と識別、いくつかのジェスチャーの検出と識別も可能になった。ジェスチャーに関しては、任意の指や手での位置からどの方向にどれだけ移動したかを検出できるようになったので、Leap Motion を用いた 1 通りの使い方に関してはプログラム可能になったと言える。批判的な部分としては、今回の手話の”どういたしまして”と”こんばんは”の検出が何度かに一度は失敗することが挙げられる。これらの原因についてはよくわからず、Leap Motion 社提供の SDK の精度に依存しているのではないかと考えている。

参考文献

- [1] 中村 薫、Leap Motion プログラミングガイド[改訂版]、I/O 編集部、株式会社工学社、東京、2015
- [2] 大澤 文孝、TWE-Lite ではじめるカンタン電子工作、I/O 編集部、株式会社工学社、東京、2014