

手指動作による文字入力インターフェース操作

Character input interface operation using hand gestures

大槻 慶人
指導教員 大島 真樹

サレジオ工業高等専門学校 情報工学科 制御情報研究室

指の簡単な動作を Leap Motion によって読み取り、文字を入力できるインターフェースを作成する。この論文ではソフトウェアの作成を行った。できている部分は画面上の文字を変えたり、指をつまむだけで文字を入力できるという機能である。

キーワード: インターフェース, 手指動作

1. 背景

画像認識やモーションキャプチャを用いて人の動作を解析する技術は進展しているが、その解析結果を活用するインターフェースの開発はまだ途上にある。また、仮想空間上で現行のキーボードを使用する場合、腕を大きく動かす必要があり、すべての人が使えるというわけではない。

このことから、腕を動かさずに指のみで効率的に文字入力ができるインターフェースを開発することを考えた。

2. 方法

Leap Motion を活用し、手指の動作を使った文字入力インターフェースを開発する。また、開発した入力手法と、現行のキーボードなど既存の入力手法について、入力速度および誤差精度を条件を変えて比較・評価する。得られた評価結果に基づき、新たに開発した入力手法に対する総合的な評価を行う。

3. 環境

本実験では下記のソフトを Windows にインストールする。

- Unity2022.3.26f1
→文字の入力画面の作成を行う。
- Visual Studio2022
→プログラムの作成を行う。
- Ultraleap Hand Tracking
→Leap Motion のハンドトラッキングを行う際に必要となる。

また、本実験では LeapMotion を使用する。

4. 作成するソフトウェア

左手を使用することを想定した場合、人差し指の中節骨側面(中指ではない方)に親指を置くことを初期状態。

親指の末節骨が、人差し指の末節骨に行くか基節骨に行くか(親指の先が右方向か左方向)で文字送りの順番を選択。

長押ししている間は文字が偏移し、指同士を離すことで文字の確定をする。

手を握ることで文字の削除とする。

文字の偏移については図1の様なソフトウェアを作成する。

手の動作については図1の様な動作とする。



図1 動作イメージ画像



図2 手の動作例

5. 方法

使用する文字に関しては日本語の50音とする。

本実験の方法としては、以下の順でプログラムを作成し、先に述べたソフトウェアを作成する。

- プログラムを作成し、指の動作(つまみ)からコンソール上に文字を表示する。
- 作成したプログラムを表示される文字が「あ」から「ん」まで順に変更されるように改良する。
- Unity3D上で入力画面の作成を行い、コンソール上に表示される文字を入力画面に表示されるようにする。
- 手の握る動作を削除として実装する。
- 指の動かす方向によって、偏移する文字を切り替える。

6. 結果

現状の結果としては、削除の実装まで行った。

使用する文字の配列は図3に記載する。

```
var word = new char[,] {
    { 'あ', 'い', 'う', 'え', 'お', 'か', 'き', 'く', 'こ', 'さ', 'し', 'す', 'せ', 'そ', 'た', 'ち', 'つ', 'て', 'と', 'な', 'に', 'ぬ', 'ね', 'の', 'は', 'ひ', 'ふ', 'へ', 'ほ', 'ま', 'み', 'む', 'め', 'も', 'や', 'り', 'ゆ', 'れ', 'よ', 'ら', 'わ', 'ん' },
}; //文字格納配列の宣言(配列名[子音][母音])
```

図3 文字の配列

図 4,図 5 は指をつまむ動作を使用してコンソール上に文字を表示し,その文字が指をつまんでいる間,偏移するようなプログラムを作成した結果である。

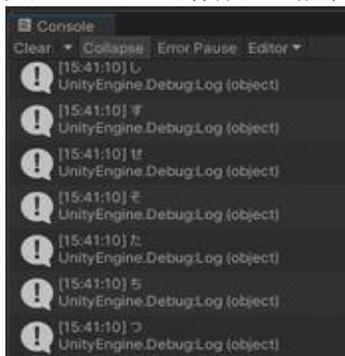


図 4 文字の偏移後

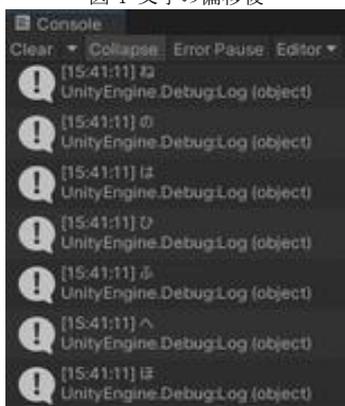


図 5 文字の偏移後

図 4,図 5 の文字を偏移させるプログラムでは文字の偏移の時間が 1 秒に 144 文字ほど偏移するものであった.その解決のため,プログラムの実行回数を数え,一定数に達したら処理を実行し,カウントをリセットするというものを作成した。

以下の表 1,表 2 は指をつまむ 1 秒ごとに偏移する文字を数えたものである.記載されている 10 の倍数はプログラムの実行回数のカウントの上限であり,0(調整前)から 90 まで 10 ずつ上限を上げた。

表 1 文字の偏移速度の調整結果(0~40)

時間	調整前	10	20	30	40
1 秒	144	14	7	5	3
2 秒	144	13	7	4	4
3 秒	144	12	7	5	3
4 秒	143	13	7	4	3
5 秒	145	12	6	4	4

表 2 調整(50~90)

	50	60	70	80	90
1 秒	144	14	7	5	3
2 秒	144	13	7	4	4
3 秒	144	12	7	5	3
4 秒	143	13	7	4	3
5 秒	145	12	6	4	4

下の図 6 は文字の表示を Unity3D の画面上で行えるようにした結果である。



図 6 Unity3D 上の文字出力

以下の図 7 は文字の削除前,図 8 は文字の削除後の図である。



図 7 文字の削除前

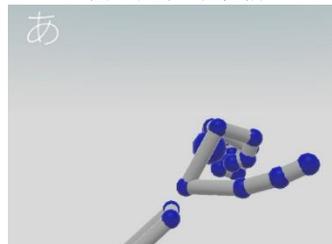


図 8 文字の削除後

9. 考察

今後開発する内容としては,文字送りの順番を「あ」から「ん」だけではなく,その逆順でもできるようにする。

- 文 献
- [1] Leap Motion を用いた空中フリック式文字入力システムの実装と評価,2018SC009 舟津春輝 2018SC098 山口晃平, 指導教員: 石原靖哲 (<https://www.st.nanzan-u.ac.jp/info/gr-thesis/2021/ishihara/18sc098.pdf>), 参照 2024-10-24
 - [2] ハンドトラッキングを用いた文字入力の VR 空間上の能力の比較, 学籍番号 18530 谷川 僚 ,サレジオ工業高等専門学校情報工学科制御情報研究室令和 4 年度卒業研究論文,2022-2023,参照 2024-10-24
 - [3] BuildInsider,Leap SDK で指を検出してみよう (Tracking Hands, Fingers, and Tools),改訂 2015-8-21, 参照 2024-10-14
 - [4] Ultraleap(<https://docs.ultraleap.com/xr-and-tabletop/xr/unity/getting-started/index.html>), 参照 2024-10-24