

Leap motion を使用した文字入力方法の提案

大西 未来^{1,a)} 渡辺 大地^{1,b)}

概要: 現在使われているソフトウェアキーボードは、タイピングが困難であることや、入力速度が物理キーボードよりもかなり劣るという問題点がある。本研究の目的は、Leap motion を入力デバイスとした新たな文字入力手法を提示し、先行手法の問題点である操作の難しさや誤認識の多さを解決することである。文字入力手法は、ジェスチャー認識とグラフィティ入力を組み合わせる方式を採用した。ジェスチャー認識は、サークルやスワイプ等の比較的に分かりやすい動作を認識する手法である。グラフィティ入力は、「グラフィティ文字」と呼ばれる軌跡と指の動作を対応することで文字入力を行う手法である。以上の手法を組み合わせ、ソフトウェアキーボードよりも直観的かつ簡素な入力方法を実現した。

Proposal of the character input method using the Leap motion

OONISHI MIRAI^{1,a)} TAICHI WATANABE^{1,b)}

1. はじめに

現在、パソコン等の電子機器に使われている文字入力を行うための一般的なデバイスはキーボードである。また、物理的なキーボードデバイスが接続されていない場合に、タッチパネルデバイスで擬似的に実現する「ソフトウェアキーボード」という技術もあり、障害者支援や銀行等の金融機関でスパイウェア対策として使われている。しかしながら、物理キーボードもタッチパネルも利用できない状況においては、ソフトウェアキーボードによる入力は困難となり、入力速度がかなり低下するという問題がある。本研究は、物理キーボードもタッチパネルも利用できない状況において、文字入力を迅速に行える手法の提案を目的とする。本研究では、この目的を達成するための手段として Leap motion と呼ばれる小型のデバイスに着目した。

Leap motion [1] とは、ユーザーの手と指の動きを感知する事でコンピューターを直感的に操作する事が可能な小型デバイスである。マウスが上下左右という平面的な操作しかできないのに対し、Leap motion では奥行きまで含めた立体的な操作が可能であり、直感的な操作が強みとしてあ

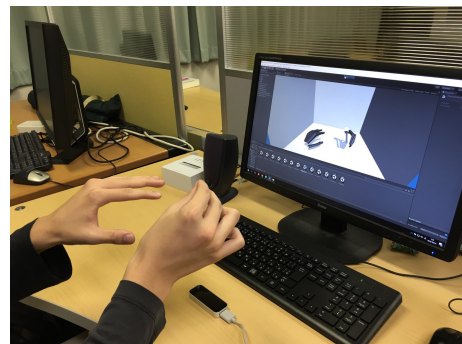


図 1 Leap Motion の使用図

Fig. 1 Image of using the Leap Motion device.

り、新たな入力デバイスとして注目を集めている。図 1 に、Leap motion を利用している様子を示す。

コンピューターの操作は勿論、ゲームや研究に多く使われており、最近では Head Mount Display (以下「HMD」) と組合わせた Virtual Reality (以下「VR」) での利用も多い。インターフェースに関連する研究においても採用事例は多く、ジェスチャー操作や文字入力という観点で多くの研究に用いられている。

本研究は Leap motion を使用し、物理キーボードがない状況での文字入力を支援することを念頭においている。利用場面の例として、アミューズメント施設などで物理キー

¹ 東京工科大学メディア学部,
Tokyo University of Technology

a) m0113081d7@edu.teu.ac.jp

b) earth@gamescience.jp

ボードが利用できない状況などがある。

Leap motion は、ジェスチャーによる入力を取得できることを特色としている。ジェスチャー検出を用いた先行研究として、藤井ら [2] による空中での手書き文字入力や、中村ら [3] による両手を使ったジェスチャー操作での入力方法などが提案されている。しかし、これらの研究では誤入力が多いことや、操作が難しいという問題がある。

本研究では、認識と操作感の問題を解決しつつ、両手を使用したジェスチャー方法を実現する。音声入力を使わず、大人子供でも使える、文字入力機能を確立し、最低でも名前入力がスムーズに出来る事を実現することが、本研究の目的である。

本研究での入力は、「ジェスチャー認識」と「グラフィティ入力」を組み合わせた手法を採用した。Leap motion は指だけでなく、それぞれの関節にも ID を指定して、動作を区別する事が可能である。

2. ジェスチャー認識

Leap motion は、デバイスの情報を取得するための開発ツールとして「Leap motion SDK」[4] (以下「SDK」) が公開されている。SDK を用いることにより、肘から指先までの形、検出した手や指の数、指先や間接の位置座標と速度ベクトル、手と各指に記された ID を検知することができる。また、ジェスチャーの種類や状態を認識する処理も行える。これらを用いることで、ピンチ処理、物をつまむといった複雑な操作の検出も可能である。

このジェスチャー認識を利用している先行研究として、細野ら [5] による提案手法がある。本研究でも、この細野らの手法を参考にして開発を行った。

片手で、指先または手全体で 4 種類のジェスチャー (サークル・キータップ・スクリーンタップ・スワイプ) を行い、認識に成功した場合は、その動きに対応した文字が出現する。図 2 に各ジェスチャー動作を示す。手首から先を回すように動かすと、種類の中にある「サークル」を認知する。回転の方向は指定していない。人差し指を下にし、勢いよく降ろす動作をすると「キータップ」といった形で文字が出てくる。スクリーンタップは他とは違い、奥行を使って認識する動作である。人差し指を真っ直ぐにし、画面をつつく様な感じで動かすと認知しやすい。指を動かす速度は、動作が遅すぎる場合は誤認識となる。自然な動作でやると認識しやすい。

このジェスチャーによって入力する情報としては、文字では無く、補助の役割をしてもらう。本手法において、SDK により認識できる 4 種類のジェスチャーのうち、いずれを用いるのが適切かについても重要な検討事項であり、これについては検証で述べる。

ジェスチャー文字入力を始める際の書き始め、そして文字の書き終わりを決定するための、役割を担ってもらうの

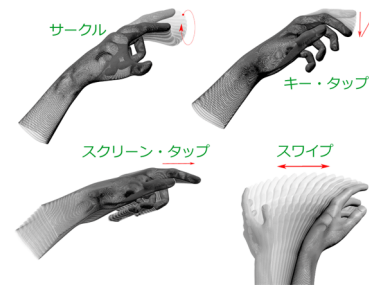


図 2 ジェスチャー例
(文献 [5] 図 4 より転載)

Fig. 2 Gesture example.

が狙いである。その上で、入力した文字を確定させる為に、グラブ (拳) を認識できるようにしておく。このグラブの動作が、キーボードで言うエンターキーの役割である。

3. グラフィティ入力

本章では「グラフィティ」と呼ばれる、手書き文字認識について述べる。グラフィティ [6] は一筆書き文字入力とも呼ばれており、元々は Palm 社が自社の PDA での文字入力方法として開発したものである。

グラフィティ文字は、アルファベットの形を簡略化したものであり、本来のアルファベット文字と比較して簡易な軌跡によって入力を可能としている。片手操作で簡単な入力を行え、スマートフォンのフリック入力よりも迅速性、確実性に優れていると言える。既に Palm 社は自社 PDA の開発は行っていないが、現在でも Android 向けの文字入力方式として利用が可能である。

図 3 はグラフィティの例である。アルファベットを崩した特徴的な形をしている。このように、それぞれのアルファベットに対応した動きを認識させる事により、文字を入力する事が出来る。通常の文字と違い、全て一筆書きで入力が完結するという特色がある。

上記にある動作を使って、ジェスチャー文字入力を行っていく。その上で重要なのが、「両手」を使うという事だ。既存研究では一筆書きや片手のみの文字入力だが、今回の研究では、上記を組み合わせかつ両手を使ったジェスチャー入力を実装する。片手でジェスチャーをし、文字の書き始め、書き終わりを指定したり、文字の決定を行う。一方で、グラフィティで文字を入力する。目標は自分や人

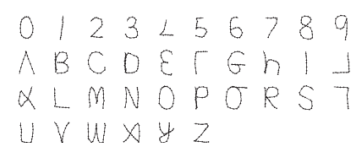


図 3 グラフィティ文字
(文献 [6] 図 4 より転載)

Fig. 3 Graffiti characters.

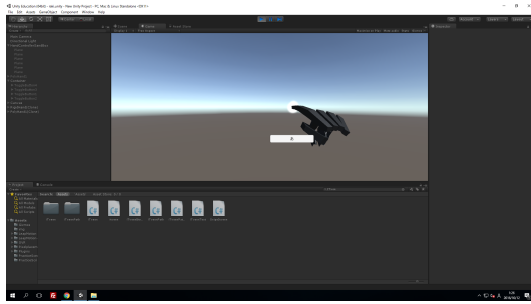


図 4 グラフィティ入力の様子

Fig. 4 Image of an input Graffiti characters.

の名前を入力してもらおうと思っている。

人差し指を伸ばすと、指先を認知してパーティクルが出現し、その後上下に指を動かし、一定の速度以上出ている場合、「あ、い」と言ったように、対応した文字が入力できる。図 4 に、グラフィティ文字による入力の様子を示す。

現状の実装では、まだ全ての文字の入力を可能とする段階には至っていない。

4. 実行結果

本手法を実装し、大学生 4 名の被験者による検証を行った。以下に、ジェスチャー入力とグラフィティ入力での検証結果を述べる。

4.1 ジェスチャー入力

ジェスチャー入力について、20 代男性の被験者 4 人に対し 4 種類のジェスチャーの入力について検証を行った。その結果、被験者の全員が 4 種類のジェスチャー認識に成功した。しかし、問題が多々見つかった。

表 1 は、誤認識も含め文字が表示されるまでにかかった総時間で、A~D は被験者を表す。規定の動作をし、しっかりと文字が出現するまで、ジェスチャーは繰り返すように指示した。なお、被験者は全員 Leap motion は未経験であった。これを踏まえ、見つかった問題点と、ジェスチャー認識に対して分かった事を記載する。

実行しそれぞれの動作をしている際、誤認識が非常に多かったという問題が発生した。比較的認識しやすい 2 つの動作である、サークル・スワイプの認識については、全員が認識に成功したが、被験者によっては入力に手間取った場合があった。しかしながら、キータップ・スクリーンタップの認識は、多くの被験者にとってかなり入力が困難である

表 1 ジェスチャー認識時間 (秒)

Table 1 Gesture recognition time

	A	B	C	D
サークル	1.50	2.34	1.43	2.02
スワイプ	1.53	1.55	1.14	1.44
キータップ	7.32	6.31	10.90	6.13
スクリーンタップ	14.6	15.50	16.88	16.28

という結果となった。キータップは、普段のキーボード操作のような指の動きでは全く反応しなかった。しっかりと指を曲げて押す、という動作をしてもらった結果、スムーズに認識できた。スクリーンタップについては、他の動作に比べて非常に時間がかかり、確実性がかなり低い結果となった。指を前方に突き出す、というスマホのタップと同じ動作で認識をするジェスチャー動作に対し、出しにくい・分かりにくいという感想が多く、多くの被験者からの口頭意見として挙がった。また、指が下を向いているときにキータップと誤認識してしまうケースも見られた。

また、4 種類同時に検知すると、他のジェスチャーとバッティングしやすく、認識精度が落ちることがわかった。そのため、本手法では認識率が高かった「サークル」と「スワイプ」の 2 種類のみをジェスチャーとして採用することとした。

4.2 グラフィティ入力

検証として、「誤認識も含め、2 種類の文字それぞれが表示されるまでにかかった総時間」を測定した。人差し指を伸ばすと、指先を認知してパーティクルが出現し、一定以上の速度で軌跡を描くことでグラフィティ入力として認識する。今回は、下方向に指を動かすと「あ」、上方向に指を動かすと「い」と認識するものとした。入力速度の検証結果を表 2 に示す。被験者は表 1 と同一である。

現時点では、まだ認識の精度は実用的と言いがたい。実験検証においては、「誤認識が酷い」、「使い辛い」、「全く文字を入力できている実感が湧かない」という口頭意見が挙がった。また、「元となる文字を書く時とは全く関係の無い動きをして、文字が表示されるのは違和感が大きかった」という口頭意見ももらっている。また、イメージできないような動きをすると、例え入力できたとしても不安に感じる事が起きてしまうという意見もあった。

5. 現状の課題と今後の展望

5.1 入力機能の向上について

Leap Motion によるジェスチャー認識は高い実用性を持つと考えられるが、本研究における現時点の状況では、まだそのポテンシャルを生かしているとは言いがたい。

現時点の重要な課題として、両手の活用がある。両手で操作している様子を図 5 に示す。

現状の実装では片手による操作のみに対応している。両手を用いる場合、ゆっくりと動かしている場合は認識に成

表 2 グラフィティ入力時間 (秒)

Table 2 Graffiti input time

	A	B	C	D
あ (下)	2.33	4.62	2.78	1.81
い (上)	1.45	2.01	1.22	2.14

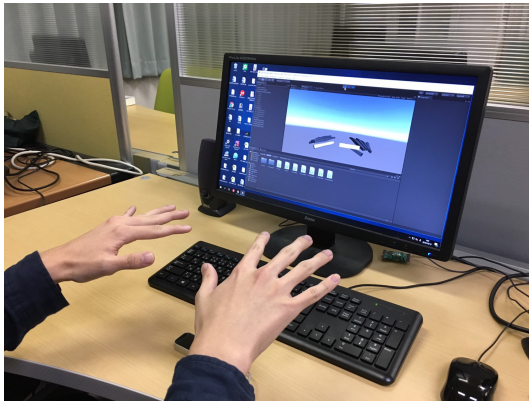


図 5 両手による文字入力
Fig. 5 Character input by both hands.

功するが、同時に激しく動かすと誤検出の割合がかなり高くなるという問題が発生した。例えば、掌が裏返る、指の1本だけが折れ曲がる等の状況を確認した。Leap motionによる認識範囲はあまり広いものではなく、ジェスチャー入力においてそれが支障となる状況が検証において多く見受けられた。この部分の改善方法を今後検討していく必要がある。

また、SDKによるジェスチャー入力において、右手と左手がそれぞれ異なった動作を行った場合、それを区別する事が出来ない。そのため、両手を有効に利用したジェスチャー入力に大きな制約が生じてしまっている。これについては、ジェスチャー認識をSDKを用いるのではなく、独自の実装で行うことで解決していく必要があると考える。

現時点では、グラフィティ手法によって入力できる文字はかなり限られている。上下左右、そしてそれらを組み合わせさせた機能を実装する。今後、アルファベットと日本語のどちらを主体にするかは検討中である。グラブの認識が出来れば、変換や大文字小文字の区別を付ける事も考えられる。ただし、多くの文字入力を可能とした場合、誤認識の割合も増大することが予想されるため、どのように対応するかについての検討も必要である。

現状では本手法においてはまだ指先のみ情報でグラフィティ文字検出を行っているが、片平ら [7] の手法のように手の中心や関節の座標を利用することで、より検出精度が高くなることが考えられる。図 6 に片平らの手法の実行の様子を示す。

実装面とは別の課題として、グラフィティ文字のデザイン面がある。文字を入力するジェスチャー動作はすぐに覚えられようような特徴的な形であることが望ましい。本手法の利用者は、ジェスチャー文字入力をする上で各文字の軌跡を覚える必要がある。実際にグラフィティ文字入力を利用した経験がない被験者からは、この軌跡を把握することが面倒だという意見もあり、今後の検討課題である。

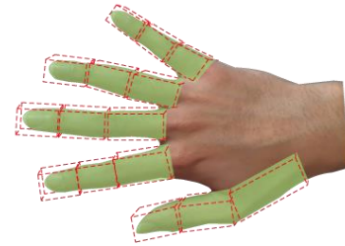


図 6 指、関節の検知例
(文献 [7] 図 2 より転載)

Fig. 6 Detection example of finger joints

5.2 応用用途

近年、HMD デバイスによる VR に関する研究 [8] や実装事例が高い注目を浴びている。実際に Leap motion を VR に適用するための開発ツールとして「Orion」 [9] 等がある。

HMD による VR コンテンツにおいて、本手法は有用なものであると考えられる。

6. まとめ

本研究では、Leap motion デバイスを用い、物理キーボードがない状況での文字入力をソフトウェアキーボードよりも迅速・確実に行える方法として、ジェスチャー認識とグラフィティ入力による手法を提案した。しかしながら、現時点ではまだ機能は限定的な段階にとどまっている。今後、本手法の完成度を高め、より実用的な入力手法の提供を目指していきたい。

参考文献

- [1] Leap Motion, Inc. Leap Motion. <https://www.leapmotion.com/>. 参照: 2016.10.14.
- [2] 藤井祐介, 竹沢恵, 真田博文, 渡辺一央. 空中手書き文字入力システムの構築に関する一考察. 研究報告モバイルコンピューティングとユビキタス通信 (MBL), Vol. 2009, No. 6, pp. 1-4, sep 2009.
- [3] 中村卓, 高橋伸, 田中二郎. Hands-Popie: 両手の動きを利用した日本語入力. *WISS2006*, pp. 151-152, 2006.
- [4] 中村薫. Leap motion プログラミングガイド [改訂版]. 工学社, 2015.
- [5] 細野敬太, 笹倉万里子, 田邊浩亨. Leap motion を用いたジェスチャー操作による文字入力方法の提案. 人工知能学会全国大会論文集, Vol. 28, pp. 1-4, 2014.
- [6] 保呂毅, 稲葉雅幸. 複数カメラを用いた手書き文字認識システム. *WISS2006*, 2006.
- [7] 片平怜士, 曾我真人. Leap Motion Controller を用いた ar 物体の把持動作表示システムの構築. 人工知能学会全国大会論文集, pp. 1-4, 2015.
- [8] 岩谷亮明, 澤田秀之. VR エンタテインメントに向けたエア楽器演奏システム. インタラクション 2014 論文集, pp. 587-592, 2014.
- [9] Leap Motion, Inc. Orion Beta. <https://developer.leapmotion.com/orion>. 参照: 2016.10.14.