

BubbleSlide：スマートウォッチ向け 円環型日本語かな入力インタフェース

東條 貴希^{1,a)} 本田 裕己¹ 加藤 恒夫¹ 山本 誠一¹

受付日 2019年2月21日, 採録日 2019年8月9日

概要：スマートウォッチ向けに各行の先頭文字を円環状に配置したフリック日本語かな入力インタフェースを提案する。テキスト入力には、小型タッチスクリーンであってもソフトキーボード領域とテキスト表示領域が必要であり、さらに日本語の場合には 50 以上のかなを入力できる必要がある。本稿では、スマートフォンの日本語入力で一般的なフリック入力を用い、フリック操作にできるだけ広い面積を提供すると同時に広い編集テキスト表示領域も確保する方式として、各行の先頭文字のキーを円環状に配置する日本語かな入力インタフェースの BubbleFlick とその改良版の BubbleSlide を提案する。BubbleFlick は、キーの 1 つにタッチすると当該行の文字が扇形に表示され、フリック操作で文字を選択する。評価実験後のインタビュー調査から、先頭文字ごとにフリック方向が異なり覚えにくいという課題が明らかになった。そこで、キーの 1 つにタッチすると当該行の文字が円環の中央に向かって一列に並ぶ BubbleSlide を開発した。提案した 2 種類のインタフェースを、スマートフォンと同じキーレイアウトの KeypadFlick キーボードと比較する 20 日間の連続評価実験を行った結果、BubbleSlide は実験最終日に 46 char/min の平均文字入力速度を示した。そしてふだんスマートフォンで KeypadFlick を使用していない実験協力者に限れば、文字入力速度とユーザビリティの主観評価スコアにおいて KeypadFlick を上回ることを確認した。

キーワード：スマートウォッチ, ソフトウェアキーボード, 日本語かな入力, BubbleFlick, BubbleSlide

BubbleSlide: Annular Layout Japanese Kana Text Entry Interface for Smartwatches

TAKAKI TOJO^{1,a)} YUKI HONDA¹ TSUNEO KATO¹ SEIICHI YAMAMOTO¹

Received: February 21, 2019, Accepted: August 9, 2019

Abstract: We developed two Japanese kana entry interfaces for smartwatches. The head letters arranged in an annular table for smartwatches. Text input requires a software keyboard with a wide space for viewing text, even on a small touchscreen. It is especially difficult to achieve this with Japanese text input because there are more than 50 kanas. The two interfaces, BubbleFlick and BubbleSlide, display the head letters around the perimeter to provide as wide an area as possible for flick operations and for viewing text. BubbleFlick radially displays four kanas from one column around the touched key, and the user selects one of the kanas by flicking. However, the flick directions were hard to memorize in a 30-day user study. To overcome this difficulty, we developed BubbleSlide, which displays four kanas from one column in a straight line from the touched key to the center. We conducted a 20-day user study comparing the two proposed interfaces with Google's standard Japanese kana entry, KeypadFlick, and BubbleSlide recorded an average text-entry speed of 46 characters per minute on the final day. BubbleSlide outperformed KeypadFlick both in text-entry speed and subjective evaluation score for the participants who do not use KeypadFlick with their smartphones.

Keywords: smartwatch, software keyboard, Japanese kana input, BubbleFlick, BubbleSlide

1. はじめに

スマートウォッチにおける文字入力は音声認識が主流で

¹ 同志社大学
Doshisha University, Kyotanabe, Kyoto 610-0394, Japan
^{a)} makuhari109@gmail.com

あるが、声を出すことができない状況ではタッチ文字入力
が便利である。しかし、スマートフォンで使用される従来
のソフトウェアキーボードをスマートウォッチの画面サイ
ズに合わせて実装すると、キーのサイズが小さくなり文字
の押し分けが困難となる Fat Finger 問題 [1] が生じる。こ
の問題を解決するために、スマートウォッチ向けに様々な
英文字の入力手法が提案されている [2], [3]。かな文字は英
文字の約 2 倍の種類があるため、英文字に適用される入力
方法では画面上の割当てが困難となる。また日本語の文は
ひらがなとカタカナ、そして漢字の組合せで構成されるた
め、インタフェースには POBox [4] のようなかな漢字変換
システムが必要である。

これまでフィーチャーフォンではテンキーの各キーに行
の先頭文字を割り当て、キーを連打することで当該行の文
字を入力するトグル入力を使用された。またその後登場
したスマートフォンでは、タッチスクリーンの特性を活
かしたフリック操作が可能となり、同じキーレイアウト
を用いたトグル入力に加えて上下左右のフリック操作で
当該行の文字を入力するインタフェース（本稿では、以降
KeypadFlick と呼ぶ）が一般的となった [5]。フリック操作
は行と段で構成されるかな文字との相性が良く、これま
でもスマートフォン上でフリック操作を利用した入力手法が
提案されている [6], [7]。

我々の研究室では、円形のスマートウォッチ向けのイン
タフェースとして、円環状に行の先頭文字を配置しフリッ
ク操作で文字を入力する黒電話フリック入力インタフェ
ースを試作した [8]。黒電話フリックの円環状に文字を並
べる設計は、キーの領域を広く確保でき、さらに画面中央
の領域に編集テキスト表示部を配置することで、多くのテキ
ストを表示できる。

本研究では円環状に行の先頭文字を配置する設計に適
した入力手法を提案する。黒電話フリックの実験から得ら
れた知見をもとに作成した BubbleFlick は、行の先頭文字
をタッチするとその周りに当該行の文字が放射状に表示さ
れ、目的の文字の方向へフリックすることで文字を入力す
る。習熟曲線を知るために実施した 30 日間の評価実験の
後に行ったインタビューから、フリックする方向が行の先
頭文字ごとに異なることが、直感的な分かりやすさを損
ねている課題が明らかになった。この課題を解決するた
めに、一定の向きのスライド操作を採用入れた BubbleSlide
を作成した。BubbleSlide は行の先頭文字をタッチすると
画面中心に向かって一列に当該行の文字が並び、目的の
文字まで指をスライドさせることで文字を入力する。ス
ライドする方向が 1 方向であるため、規則的に操作する
ことができる。BubbleFlick と KeypadFlick を比較対象
とし、20 日間の比較実験を実施した。本稿ではこれら
について報告する。

2. 関連研究

これまで、小型端末向けのソフトウェアキーボードとし
て英文字の入力手法が多く提案されている。英文字とか
な文字は文字数や構成が異なるが、Fat Finger 問題のよ
うに共通する課題も多くある。

2.1 スマートウォッチ向け英文字入力方式

スマートウォッチのような小さな画面では、文字のキー
領域が狭くなり、Fat Finger 問題が生じる。この問題を
解決するための英文字の入力手法が多く提案されている。

小さな QWERTY キーボードを適応的に拡大表示しな
がら入力する手法が提案されている [2], [9], [10], [11]。
ZoomBoard [2] はキーボードの大きかな位置に対するタ
ップ操作により一部を段階的に拡大した後、キーに対
するタップにより入力文字を確定する。6 名の実験協
力者に対して評価した結果、文字入力速度は 9.3 word/min
を示したと報告している。ZShift [9] は、選択中の文字
とその周辺の領域を画面上部のテキスト表示部に拡大
してポップアップ表示する手法を用いることで、選
択中の文字が指で隠れてしまう問題を解決している。
文字を選択している状態から、ポップアップを見なが
ら指を動かすことでその周辺の文字を選択できる。
Flickey [10] は、ZShift のポップアップの機能を採
り入れ、タッチした列が目的の列と異なる場合には
そのまま左右にドラッグすることで列を選択し直し、
そのまま指を離すか上下にフリックすることで文字
を入力する。DriftBoard [11] はキーボードの文字を
タッチするのではなく、画面上の小さなキーボード
自体を指で動かし、所定の位置に目的の文字をあ
わせることで入力する。

キーボードの分割表示や 1 つのキーへの複数文字の
割り当てにより表示するキーの数を減らし、1 つ 1
つのキーを大きくする手法がいくつか提案されてい
る [12], [13], [14], [15]。SplitBoard [12] は
画面上に QWERTY キーボードの半分の領域を
表示し、左右のスワイプにより切り替える。
Swipeboard [13] は QWERTY 配列の文字を 9
つのキーに分けて割り当て、最初のスワイプで
キーを選択し、2 回目のスワイプで文字を
特定する。ETAO Keyboard [14] はメ
イン画面に最もよく使用する 8 文字を並べ、
残りの文字は英文字順に 3 つの画面に分けて
表示する。文字を入力するたびにメ
イン画面に戻る。5-TILES-Keyboard [15] は、
英文字を画面に横並びになった 5 つの
キーに割り当て、横スライドによる
操作で文字を特定する。

ここまで紹介した手法は、タッチ位置のズームや画面の
切り替え等、複数の段階を踏む必要がある。これに
対して、単語辞書や言語モデルを利用して確率の
高い単語を予測したり、自動修正を行ったりする
統計的デコーディング手法が提案されている [3],
[16], [17]。Komninos ら [16] は英
文字を 6 つに削減したキーに分けて割り
当て、タップされた

キーの系列から確率の高い単語を予測する。VelociTap [17] は、確率的キーボード、文字レベルの言語モデル、単語レベルの言語モデルを組み合わせて、スペースの挿入を省略できる文単位のデコーディングを提案した。40mm 幅のキーボードを用いて 41 word/min の入力速度を報告している。WatchWriter [3] はタッチ用の確率的キーボードと 1 ストロークで単語を入力できるジェスチャキーボードを組み合わせ、22–24 word/min の入力速度を報告している。

2.2 円環型英文字入力方式

英文字の入力では、使用するデバイスによっては円環状に文字を配置する手法が効率的に入力できると主張されている [18], [19]。スタイラスペンを使用して入力する Cirrin [18] は円環状に並んだ文字をジェスチャーにより連続して入力する。卓上ディスプレイに実装された Bubble-CIRCLE [19] はタッチ入力とジェスチャー入力に対応しており、次に入力される可能性があるキーのサイズを動的に変化させる。

スマートウォッチの小さな画面上に 26 文字の英文字を円環状に並べるとキーがかなり小さくなるが、この問題を解決する手法がいくつか提案されている [20], [21], [22]。WatchMI [20] は文字を 8 つのキーに分け、キーを選択してからデバイスを傾ける大きさによって文字を選択する。HARI キーボード [21] は環状に並んだ子音の英文字をタッチしてから画面中央部まで指をスライドさせると子音の文字が母音に切り替わり、目的の文字にフリックすることで入力する。TouchOne Keyboard [22] は単語辞書を導入し、環状に並んだ 3~4 の英文字を 1 つのキーとすることで文字を入力する。また TouchOne Keyboard には単語辞書に依存しない入力方法もあり、タッチしたキーの周りにグループ内の文字が表示され、目的の文字の方向にフリックすることで入力できる。

2.3 スマートウォッチ向けかな文字入力方式

かな文字は子音と母音で構成されるため、フリック操作やスライド操作と組み合わせた操作が適している。この特徴を利用した手法がいくつか提案されている [23], [24], [25]。ShuttleBoard [23] は 10 個の子音キーと 5 個の母音キーで構成されており、子音キーから母音キーへ指でなぞることで文字を入力する。また指を離すことなくこの動作を繰り返すことで連続して文字を入力できる。24 名の実験協力者を対象に 2 日間評価した結果、最終セッションで文字入力速度が 49.2 char/min を示したと報告している。ShuttleBoard はタッチアップせずに文字入力できる代わりにキーの数が 15 個と多く、入力部の画面占有率が高い。秋田ら [24]、梅澤ら [25] はスマートウォッチの小さな画面において誤入力率を低く抑えたまま、入力部の画面占有率を小さくする入力手法を提案した。秋田らは矩形の画面の縁

に沿った 2mm の領域にキーを配置し、画面外側から内側に向けて指をスライドさせることで確実に入力を検知できる手法を提案した。提案手法の入力部の画面占有率は 1.6 インチの正方形画面で 26.4% に抑えたと報告している。梅澤らは画面上下の縁に沿って表示領域が 1×5mm、検知領域が 3×6mm のキーを 12 個配置し、2 段階の操作で入力する手法を提案した。提案手法の画面占有率は 9.1%、誤入力率は 2.17% を示したと報告している。

3. BubbleFlick の設計と試作

スマートウォッチの小さな画面上でフリック操作のためにできるだけ広い面積を提供すると同時に十分な広さの編集テキスト表示部も確保するため、円環状に各行の先頭文字を配置する BubbleFlick を設計した。図 1 に BubbleFlick のスクリーンショットを示す。インタフェースの特徴を 1)~5) に示す。

- 1) 直径 35mm の円形タッチディスプレイ上に、円環状に検知領域が 7mm 四方の行の先頭文字のキーをキー中心の間隔を 7mm として配置した。キーの中央に表示する文字の大きさは 4mm 四方とした。キーをタッチするとそのキーを始点として放射状に当該行の文字が表示される。タッチした状態から目的の文字に向けてフリックすることで文字を入力する。入力部の内側には縦 16mm、横 14mm の編集テキスト表示部を配置した。この画面レイアウトは、文字サイズを大きく保ちながら隣の文字との誤タッチを防ぐことができ、また編集テキスト表示部の領域を広く確保できる。これにより、テキスト表示部にはかな文字を最大で 49 文字表示できる。すべてのキーと編集テキスト表示部は画面サイズに対して相対的にサイズと配置が決められる。矩形のディスプレイの場合も円環状に行の先頭文字のキーが表示される。
- 2) 選択された文字を見やすくするために、文字に背景色をつけた。操作中は編集テキスト表示部と選択文字以外の文字が半透明になるようにした。
- 3) 画面中央の編集テキスト表示部における、改行の挿入、文字の削除、カーソルの左右移動、入力の確定の 5 種類の機能をそれぞれ、円環の内側の領域における上下左右のフリックとロングタップのタッチ操作に割り当てた。これにより、画面を構成するボタンの数が少なくなり、編集テキスト表示部を広く確保できる。タッチ操作と機能の対応関係を表 1 に示す。
- 4) 選択中の文字のポップアップを指で隠れない位置に表示し、文字を確認できるようにした。
- 5) 右上の濁点のキーを選択することでカーソル位置の文字を濁音、半濁音、捨て仮名に切り替えられるようにした（以降、文字変換と呼ぶ）。また文字に対して切り替え先が 2 つ以上存在する場合は、キーの連打によ

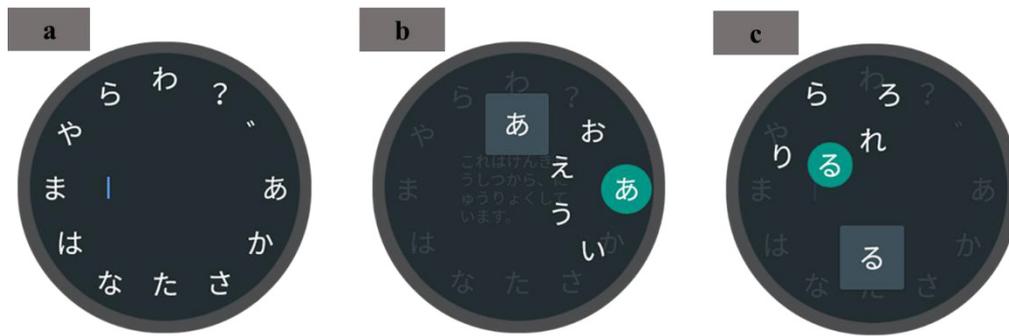


図 1 BubbleFlick. a) 初期画面, b) 「あ」をタッチした場合, c) 「る」をフリックした場合 (実物大)

Fig. 1 Screenshots of BubbleFlick interface, a) initial screen, b) when button 'a' is touched, and c) when button 'ru' is flicked.

表 1 BubbleFlick におけるタッチ操作と対応機能

Table 1 Touch operations and corresponding functions in BubbleFlick.

タッチ操作	対応機能
上フリック	改行の挿入
下フリック	1文字の削除
左右フリック	カーソルの移動
ロングタップ	入力の確定

り順番に切り替わるようにした。

4. プロトタイプ評価実験

4.1 実験の概要

BubbleFlick の性能と操作性を検証するために評価実験を実施した [26], [27]. インタフェースに対する習熟曲線を知るため実験期間は 30 日間とした. 実験協力者は 22~25 歳の合計 8 名とした. 男性 7 名, 女性 1 名, 右利きが 7 名, 左利きが 1 名であった. スマートフォン歴は 4~8 年であり, 全員がスマートウォッチを使用したことがなかった.

実験協力者には毎日自宅等の静かな環境の中, BubbleFlick と 5.2 節で後述する比較用の KeypadFlick のそれぞれのインタフェースで, 日替わりの文セットを入力してもらった. 使用するインタフェースの順番が結果に影響しないように, 日ごとに入力するインタフェースの順番が交互になるようにした. 実験終了後に 30 日間の実験に対する取り組み方とインタフェースの良かった点と悪かった点をインタビュー形式で回答してもらった.

5 文もしくは 6 文の短文を 1 セットとし, 1 セットの短文の文字を足し合わせるとすべてのひらがなが 1 回以上使用されるようにしている. 1 セット合計の文字数は実験協力者の負担を考慮して, 約 100 文字とした. 実験協力者が参照する用紙には, 1 日分の短文をかな文字とそのすぐ隣に漢字かな交じりの文として印刷した. 実際に使用した短文セットを構成するかな文字の文例を表 2 に示す.

文字入力速度, 文字入力精度は計測している [26], [27] が,

表 2 実験で使用した短文セットの例

Table 2 Sample Kana sentences used in experiment.

- ① わたしはどうししゃだいがくのがくせいです。
- ② ひとり、ふねにのってむかう。
- ③ そこぬけにあかるいひよこ。
- ④ ゆきのなか、つうろをあるくちまみれのおんな。
- ⑤ かっぱはえさをもとめて、ほらあなへやってきた。

本稿では結果の再掲は省略し, フリック操作インタフェースの改良につながるインタビュー調査結果のみ記す.

4.2 インタビュー調査

30 日間の実験終了後に実施した BubbleFlick についてのインタビューの結果を示す. ほとんどの実験協力者から, 文字の打ちやすさ, 入力されたテキストの見やすさに対する高評価が得られた. しかしながら, 3 章. の 1) のフリック入力の操作性に関して, 「フリック先の文字の並びの規則性が分からない」, 「ま行, や行以降からのフリック先の文字の並びが覚えられない」という意見が得られた. フリック方向の規則性に対して 30 日間で完全に覚えたとは回答した実験協力者はいなかった. BubbleFlick では円環状に並んだ行の先頭文字ごとにフリックする方向が異なるため, 直感的に操作することが難しいと推察された. この問題を解決できれば, 文字入力速度の向上が期待できる.

編集テキスト表示部については, 文字の大きさ, 領域の広さに対して予想どおりの好評価が得られた. 一方で, 表 1 に示したタッチ操作について「フリックの方向を間違えてしまう」, 「繰り返し同じ操作をする場合には時間がかかる」といった意見が得られ, 改善が求められた.

3 章の 4) のポップアップの表示は, 実験協力者のほとんどが見えていなかった. ポップアップよりもフリック方向を画面上で確認することに注意が向けられていたと考えられる.

インタビューに対する回答から, BubbleFlick のさらなる操作性向上につながる以下の課題が明らかになった.

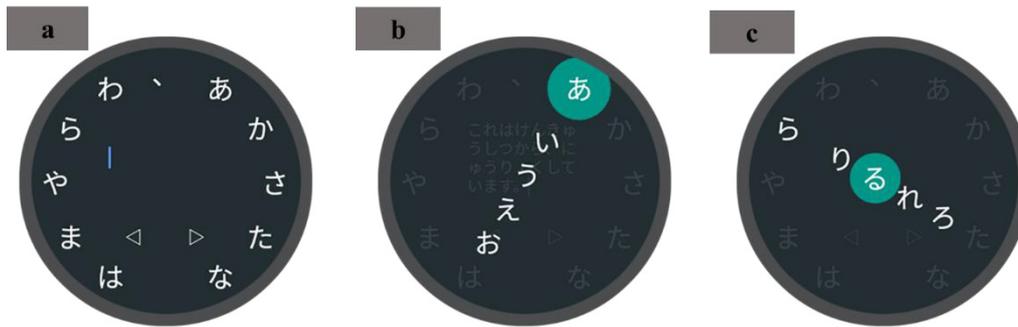


図 2 BubbleSlide. a) 初期画面, b) 「あ」をタッチした場合, c) 「る」までスライドした場合 (実物大)

Fig. 2 Screenshots of BubbleSlide interface, a) initial screen, b) when button ‘a’ is touched, and c) when sliding onto button ‘ru’.

- I. 円環状に並んだ行の先頭文字ごとにフリックする方向が異なるため、直感的に覚えることが難しく、フリック方向の文字を確認することなく機械的にフリックすることが難しい。
- II. 編集テキスト表示部内のタッチ操作と対応する機能の組合せが適切でない。具体的には、カーソル移動や文字の削除が1文字ずつであると、一度に操作する回数が多い場合には時間がかかってしまう。また文字の削除と改行の挿入に対して、フリックする方向を間違えてしまう。

5. フリック操作の規則性を高めた円環型かな文字入力インタフェース

5.1 BubbleSlide の設計と試作

前節の実験で明らかになった BubbleFlick におけるフリック方向の規則性の課題に対応することで、新たに BubbleSlide を作成した。図 2 に BubbleSlide のスクリーンショットを示す。以下に課題としてあげられた前節の I, II に対する改修内容と、新たに改良した機能の内容 III, IV, V を示す。

- I. 円環状に並んだ行の先頭文字を選択すると画面中心に向かって一列に当該行の文字が並ぶようにした。文字の入力は目的の文字まで指をスライドさせることで行う。これにより、指を動かす方向は画面中心への1方向に定まるため、規則的にスライドすることができる。一列に並んだキーの間隔は‘あ段’から‘い段’までキー中心間の距離を8mmとし、それ以外を5mmとした。スライドによる文字入力の特徴として、‘あ段’はタップのみ、‘い段’はタッチしてから少し動かす、‘お段’は画面の端まで指でなぞることで入力ができる。一方で、‘う段’と‘え段’ではこれらのような特徴がないため、文字の押し分けが難しいと考える。そのため‘う段’と‘え段’の文字の境界を通過する際に振動フィードバックを与えるようにした。

表 3 BubbleSlide におけるタッチ操作と対応機能

Table 3 Touch operations and corresponding functions in BubbleSlide.

タッチ操作	対応機能
ダブルタップ	改行の挿入
下フリック	1文字の削除
左右フリック	濁点, 半濁点, 捨て仮名の入力
ロングタップ	入力の確定

- II. カーソルの左右の移動は、編集テキスト表示部の下に配置したカーソルキーで操作できるようにした。カーソルキーを増やしたことで、編集テキスト表示部は縦11mm, 横14mmとなった。テキスト表示部にはかな文字を最大で35文字表示できる。対応する機能がなくなった左右フリックには文字変換を割り当て、左右どちらにフリックしても同じように切り替わるようにした。文字の削除と改行の挿入に対するフリック方向の間違いを防ぐために、改行をダブルタップに割り当てた。BubbleSlide におけるタッチ操作と機能の対応関係を表 3 に示す。
- III. 文字変換を左右フリックのタッチ操作で行うことにより、円環状に並んだ文字のキーから文字変換のキーをなくせる。そのため、空いた領域を詰めるようにして行の先頭文字のキーを再配置し、行の先頭文字が画面の左右に分かれて並ぶようにした。
- IV. キーの表示領域を検知領域よりも小さくした場合に誤入力率が低減されるため [28], 行の先頭文字のキーの検知領域を7mm四方から8mm四方に拡大した。
- V. BubbleFlick は文字を記したポップアップを表示していたが、ポップアップを見ないというインタビューの回答が多く得られたため、表示をなくした。

5.2 テンキーフリック式かな入力インタフェース

比較用の KeypadFlick はスマートフォン等で広く利用されている日本語入力インタフェースである。図 3 に 2017



図 3 KeypadFlick. a) 初期画面, b) 「な」タッチ時

Fig. 3 Google's Japanese text-entry method on a smartwatch. a) initial screen, and b) when button 'na' is touched.

年に Google がリリースした Android スマートウォッチ向け日本語かな入力インタフェースのスクリーンショットを示す。画面上部には縦 4 mm, 横 15 mm の編集テキスト表示。0 部, それより下部は入力部になっている。入力部の内側には 2 mm 四方で行の先頭文字, 濁点半濁点, 句読点のキーが, キー中心の間隔が縦 3 mm, 横 7 mm で配置されている。入力部の外側にはカーソル移動, 文字の削除等のキーが並ぶ。文字のキーをタッチすると上下左右に当該行の文字が表示され, 目的の文字の方向にフリックすることで文字を入力する。また予測変換機能により, 入力部の上部に候補文字が表示される。入力方式にはフリック入力に加えてトグル入力を採用しており, どちらでも入力ができる。トグル入力では文字のキーを連続してタッチすることで, 次の行の文字を入力する。

今回の実験では, かな文字入力機能に限定して比較評価するため, 実験協力者には KeypadFlick が備える予測変換機能等のかな文字入力以外の機能は使用しないように伝えた。

6. ユーザ評価実験

6.1 実験の概要

BubbleSlide の性能と操作性を検証するために BubbleFlick と KeypadFlick を比較対象とし, 比較実験を実施した。同じ実験協力者が複数のインタフェースを使用すると Skill Transfer [29] が起こる可能性があるため, 実験協力者を, BubbleFlick を使用するグループ, BubbleSlide を使用するグループ, KeypadFlick を使用するグループに分けて評価した。実験期間は 20 日間とし, 毎日自宅等の静かな環境で割り当てられたインタフェースを用いて用意した短文のセットを入力してもらった。

6.2 評価条件

6.2.1 実験協力者

実験協力者は, BubbleFlick の実験に関与していない 21~24 歳の大学生 33 名とした。男性が 24 名, 女性が 9 名であり, 右利きが 30 名, 左利きが 3 名である。すべての実験協力者が日頃からスマートフォンを使用しており, スマ

ートフォン使用歴は 3~8 年であった。そして実験協力者のうち 3 名は, スマートウォッチ使用歴が半年から 1 年未満であった。それ以外の実験協力者はスマートウォッチを使用したことがなかった。

6.2.2 使用したデバイス

本実験では Moto360 2nd Gen を使用した。Moto360 2nd Gen の画面解像度は 263 ppi, 画面の直径は 34.8 mm である。

6.2.3 用意した短文

18 文の短文を 1 セットとする日替わりの文セットを用意し, 実験協力者には 1 日に 1 セットを入力してもらった。1 セットの文字数は約 300 文字とし, 文字を足し合わせると, すべてのひらがなが 1 回以上使用されるようにした。実験協力者が参照する用紙には, 1 日分の短文をかな文字とそのすぐ隣に漢字かな交じりの文として印刷した。また, 短文の順番が結果に影響を与える可能性があるため, 実験協力者ごとにセット内での短文をランダムに並べた。

6.2.4 評価指標

性能評価指標は文字入力速度 (CPM, char/min) と誤入力率 (EPC, error/char) とした。計測は全文まとめて, デバイス内で自動的に行うようにした。インタフェースの起動と同時に計測を開始し, 入力の確定が押された時点で計測を終了するようにした。文字入力速度は計測開始時から計測終了時までに入力されたかな文字の合計数を計測時間で割って求める。誤入力の文字数は合計数に含まれ, 削除キーの入力回数は合計数に含まれていない。

ユーザビリティの主観評価を取得するために, 実験終了後には SUS (System Usability Scale) によるアンケートを実施した。SUS はシステムのユーザビリティを 0~100 点で評価し, 点数が高いほどユーザビリティは高くなり, 使いやすいシステムとされる。約 2300 のシステムに対する SUS の統計的分析により, SUS の評価による一般システムの平均点は 70 点とされている [30]。SUS の原文は英語であるため, 古井らの実験 [31] で使用されている日本語訳版を使用した。

6.3 実験の手順

実験では Skill Transfer を回避するために, BubbleFlick を使用する実験協力者を 11 名, BubbleSlide を使用する実験協力者を 11 名, KeypadFlick を使用する実験協力者を 11 名に分けて評価した。加えて KeypadFlick を使用する実験協力者のなかでは, ふだんからスマートフォンで KeypadFlick を利用している 5 名を KeypadFlick ユーザグループとし, KeypadFlick を利用したことがない, またはフリック操作に慣れていない 6 名を KeypadFlick 非ユーザグループとした。スマートウォッチの使用経験がある実験協力者 3 名は, 1 名が BubbleFlick, 2 名が KeypadFlick ユーザグループに割り当てられている。

実験協力者には, 他者との接触がない静かな環境で, 日ご

とに用意した短文のセットを割り当てられたインタフェースで入力してもらった。また決まった時間帯にその日の実験を開始してもらった。入力する姿勢は自身が最適に入力できる姿勢に定め、デバイスを利き腕と反対側の手首に装着してもらい、入力する指を人差し指に限定した。実験開始時には実験協力者に対して実験の手順を説明し、インタフェースの操作方法を理解してもらうために練習用の5つの短文を入力してもらった。

6.4 評価結果

6.4.1 文字入力速度

実験協力者ごとに20日間の回帰曲線を求め、その推定値からインタフェースごとの平均文字入力速度を算出した。回帰曲線のモデル式を式(1)に示す。モデル式は11種類の曲線回帰モデルのうち最も決定係数が高くなった式を採用した。式において、 t は日数、 b_0 、 b_1 は係数である。

$$y = b_0 t^{b_1} \quad (1)$$

日ごとのグループ平均の文字入力速度を棒グラフ、回帰曲線を折れ線グラフとして図4に示す。図4よりすべてのインタフェースにおいて、20日間による操作の習熟が表れた。BubbleSlideとBubbleFlickを比較した場合、開始当初はほとんど同じ値を示したが、日が経つにつれてBubbleSlideの方が高い値を示した。KeypadFlickのグループ間の比較では、KeypadFlick ユーザグループの方が20日間を通して高い値を示した。KeypadFlick ユーザグループがスマートフォンで覚えたKeypadFlickへの習熟がスマートウォッチでも表れたといえる。

次に、図4から習熟による変化を見るために回帰曲線から算出した1日目から約10日ごとの推定値を表4に示す。BubbleSlideの実験1日目は28.0 char/min、実験20日目は46.2 char/minを示し、20日間の変化率として見ると1分間に約1.7倍多く打てるようになった。改良前後の比較では、BubbleSlideの実験20日目の46.2 char/minと、BubbleFlickの実験20日目の42.6 char/minを比較して、1分間に約1.1倍多く入力できるようになった。図4におけるBubbleSlideとKeypadFlickの20日間の値を比較した場合、BubbleSlideよりもKeypadFlick ユーザグループの方が20日間を通して平均で7%高かったが、KeypadFlick 非ユーザグループに対してはBubbleSlideの方が平均で10%高かった。

インタフェースの間の20日間の平均文字入力速度の有意差を調べた。まず、実験協力者全体の回帰曲線を求め、その曲線と実験協力者の回帰曲線との日ごとの誤差から平均誤差を算出し、インタフェースごとに平均値を求めた。平均誤差の値を表5に示す。平均誤差に対して、インタフェースを要因とする一要因分散分析を行った。分散分析の結果、有意な群間差が見られた ($F_{(3,594)} = 17.08, p < 0.01$)。

表4 回帰曲線から算出した日数ごとの平均文字入力速度

Table 4 Estimated text-entry speeds (CPMs) for each interface based on regression curves averaged.

インタフェース	1日目	10日目	20日目
BubbleFlick	27.3	38.4	42.6
BubbleSlide	28.0	41.1	46.2
KeypadFlick ユーザグループ	30.7	44.1	49.3
KeypadFlick 非ユーザグループ	22.0	37.9	44.7

表5 インタフェースごとの回帰曲線と全体の回帰曲線との平均誤差

Table 5 The mean errors between the regression curve for each interface and the overall regression curve.

インタフェース	平均誤差	標準偏差
BubbleFlick	-0.9	8.5
BubbleSlide	1.5	6.0
KeypadFlick ユーザグループ	4.4	6.5
KeypadFlick 非ユーザグループ	-1.4	5.6

TukeyのHSD法による多重比較(5%水準)を行った結果、BubbleSlideはKeypadFlick ユーザグループに対して有意に低い値を示した($p < 0.01$)。一方で、BubbleFlickとKeypadFlick 非ユーザグループに対しては有意に高い値を示した($p < 0.01$)。BubbleSlideがBubbleFlickよりも文字入力速度が高くなった要因として、5.1節のI.による改良により、直感的に入力できるようになったためであると考える。実験終了後に文字の配置と並びは覚えたかと質問した結果、BubbleFlickではすべて覚えたかと答えた協力者は1人もいなかったのに対し、BubbleSlideでは半数以上の協力者がすべて覚えたかと回答した。またすべて覚えたかと回答した協力者からは、「慣れてくるとどれくらい指を動かせばいいか感覚で分かるようになった」、「文字の配置や並びが単純だった」というコメントが得られた。BubbleSlideの文字入力速度はKeypadFlick ユーザグループよりも低く、KeypadFlick 非ユーザグループよりも高くなった。これより、インタフェースに対する事前の習熟がなければ、BubbleSlideの方が速く文字を入力できることが明らかになった。

6.4.2 誤入力率

文字入力速度と同様に、日ごとに平均誤入力率を算出した。KeypadFlickはユーザグループと非ユーザグループともに実験開始当初の値が高く、時間の経過とともに低下したが、BubbleFlickとBubbleSlideは1日目から低い値を示し、20日間を通して安定して低かった。インタフェースへの習熟が進んだ段階での誤入力率を比較するため、実験開始当初のKeypadFlickの不安定な誤入力率を除き、実験16~20日目の最後の5日間について算出した誤入力率を表6に示す。表6の平均誤入力率に対して、インタフェースを要因とする一要因分散分析を行った。分散分析の結果、有意な群間差が見られた ($F_{(3,177)} = 5.45, p < 0.05$)。Tukey

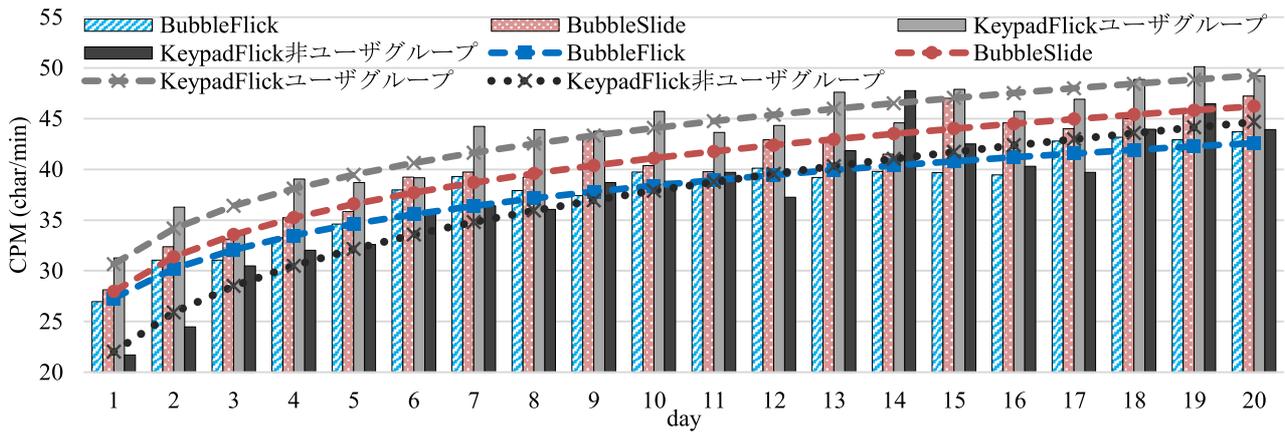


図 4 回帰曲線から算出したインタフェースごとの 20 日間の平均文字入力速度

Fig. 4 Daily estimated text-entry speeds (CPMs) for each interface based on regression curves averaged.

表 6 インタフェースごとの最後 5 日間の平均誤入力率

Table 6 Last 5 days of error rates averaged for each interface

インタフェース	平均誤入力率 (error/char, %)	標準偏差
BubbleFlick	7%	0.03
BubbleSlide	8%	0.04
KeypadFlick ユーザグループ	8%	0.03
KeypadFlick 非ユーザグループ	10%	0.07

表 7 インタフェースごとの SUS の平均点と標準偏差

Table 7 Mean and standard deviation of System Usability Scale (SUS) scores for each interface.

インタフェース	平均点	標準偏差
BubbleFlick	68	16
BubbleSlide	78	11
KeypadFlick ユーザグループ	61	7.6
KeypadFlick 非ユーザグループ	52	9.8

の HSD 法による多重比較 (5%水準) を行った結果, BubbleFlick の 7%は KeypadFlick 非ユーザグループの 10%に対して有意に低い値を示した ($p < 0.05$). BubbleSlide と他のインタフェースの間には有意差は見られなかった. 20 日間の変化を見ると BubbleFlick と BubbleSlide は 1 日目から低い値を示し, 20 日間を通して値の変動が小さいことから安定した入力できていた. KeypadFlick ユーザグループと KeypadFlick 非ユーザグループは実験開始当初 BubbleSlide よりも平均誤入力率は高かったが, 最後 5 日間の比較ではその差はなくなった. これより, KeypadFlick は操作性に慣れていない最初の段階では誤入力を頻繁に引き起こすが, 慣れてくるにつれて BubbleSlide と同程度の誤入力率まで減少する.

6.4.3 ユーザビリティ評価

20 日間の実験終了後に, 使用したインタフェースについて SUS のアンケートに回答してもらった. インタフェースごとの SUS の平均点と標準偏差を表 7 に示す. BubbleSlide の 78 点は一般システムの平均点である 70 点を上回り, かつ 3 つのインタフェースのなかで最も高い点数を示した. SUS の平均点に対して, インタフェースを要因とする一要因分散分析を行った. 分散分析の結果, 有意な群間差が見られた ($F_{(3, 29)} = 6.37, p < 0.05$). Tukey の HSD 法による多重比較 (5%水準) を行った結果, BubbleSlide は KeypadFlick 非ユーザグループの 52 点に対して, 有意に

高い値を示した ($p < 0.05$). しかし, BubbleFlick の 68 点と KeypadFlick ユーザグループの 61 点の間では, 有意差は見られなかった.

BubbleFlick と BubbleSlide の SUS の値は個人差が大きかった. BubbleFlick と BubbleSlide は一般的に使用されているソフトウェアキーボードにはない特徴的なレイアウトと操作方法になっている. 実験協力者がそこに違和感を覚えたかどうかで SUS の値に大きな個人差が生じたのではないかと考える. KeypadFlick 非ユーザグループの点数が最も低かったのは, キーのサイズが小さいうえ, フリック操作に慣れていなかったために誤入力が頻繁に起きたことが原因と考えられる. 一方で, BubbleSlide の点数が最も高かったのは, 1 方向に指をスライドさせるだけですべての文字を入力できるため, 初めて操作する場合でも簡単に入力でき, かつ誤入力が少なかったためだと考えられる.

BubbleSlide はいまだ KeypadFlick ユーザグループに文字入力速度で勝っていないが, 使いやすさの点では勝っているため, 使い続けるうちに文字入力速度もさらに改善されることが期待される.

今回の評価実験では, 直径約 35 mm の円形ディスプレイのスマートウォッチを使用した. 未検証ではあるが, 一般的なサイズの円形および矩形ディスプレイのスマートウォッチにおける使いやすさについて考察する. まず, Google が提供するスマートウォッチのエミュレータ [32] の画面サイズを調べると, 円形タイプで直径が 30 mm から 42 mm, 矩

形タイプで縦 23 mm, 横 23 mm から縦 30 mm, 横 30 mm までが用意されている。円形タイプにおいて直径 35 mm は平均的なサイズといえる。画面サイズがそれより小さい場合, BubbleSlide では画面サイズに相対的にキーのサイズを縮小することで中央の編集テキスト表示部は相応に狭くなるが, 直径 30 mm まで実用的な操作性を提供できると考えている。矩形タイプの場合, 最大級の縦 30 mm, 横 30 mm であればろうじて対応できるが, それよりも小さくなるとキーのサイズと間隔は小さく, 編集テキスト表示部の文字数も厳しく制限されてしまうと予想される。KeypadFlick の場合は, 直径 35 mm の円形ディスプレイにおいてキーのサイズ, 間隔ともにすでに小さいため, ディスプレイサイズに対する条件はより厳しいと考えられる。

7. おわりに

本研究では, スマートウォッチ向けに円環状に行の先頭文字を並べるキーレイアウトとフリック操作を組み合わせたかな文字入力手法を提案した。フリックの素早い操作性を追求して作成した BubbleFlick は 30 日間の評価実験からフリック先の文字の並びが直感的に分かりにくいという課題が明らかになった。この課題を解決するために, フリック操作の規則性を高めた BubbleSlide を作成した。作成したインタフェースの性能は, スマートフォンで一般的な KeypadFlick を比較対象として 20 日間評価した。

BubbleSlide と BubbleFlick の改良前後の比較では, 改良の結果, 実験 20 日目には 1 分間に 1.1 倍多く入力できるようになった。またユーザビリティにおいても, 改良後の SUS の平均点は一般システムの平均点を大きく上回る結果となった。BubbleSlide を KeypadFlick と比較すると, KeypadFlick ユーザグループの文字入力速度を上回することはできなかった。KeypadFlick ではスマートフォン等での慣れが入力速度に大きく影響しており, 慣れてきたユーザにとっては KeypadFlick が最も素早く入力できた。しかし, KeypadFlick 非ユーザグループとの比較では, 20 日間を通して BubbleSlide の方が平均で 10% 高くなった。さらにユーザビリティにおいても BubbleSlide の方が高かった。

比較実験から, BubbleSlide は初心者でも直感的に操作できるインタフェースであることが明らかになった。今後の展望として, BubbleSlide を実用的なものにするために, かな漢字変換機能と予測変換機能の導入を検討している。

参考文献

- [1] Siek, K.A., Rogers, Y. and Connelly, K.H.: Fat Finger Worries: How Older and Younger Users Physically Interact with PDAs, *Proc. INTERACT'05*, pp.267–280 (2005).
- [2] Oney, S., Harrison, C., Ogan, A. and Wiese, J.: ZoomBoard: A Diminutive Qwerty Soft Keyboard using Iterative Zooming for Ultra-small Devices, *Proc. CHI'13*, pp.2799–2802, ACM (2013).
- [3] Gordon, M., Ouyang, T. and Zhai, S.: WatchWriter: Tap and Gesture Typing on a Smartwatch Miniature Keyboard with Statistical Decoding, *Proc. CHI'16*, pp.3817–3821, ACM (2016).
- [4] Masui, T.: POBox: An Efficient Text Input Method for Handheld and Ubiquitous Computers, *Proc. HUC1999*, pp.289–300 (1999).
- [5] Wikipedia: Japanese Input Method, Wikipedia (online) available from (https://en.wikipedia.org/wiki/Japanese_input_methods) (accessed 2019-05-28).
- [6] 青木良介, 橋本 遼, 瀬古俊一, 片岡泰之, 井原雅行, 渡辺昌洋, 小林 透: Drag&Flick: タッチスクリーンを用いた視覚障がい者向け文字入力方式, インタラクシオン 2013, pp.72–79, 情報処理学会 (2013).
- [7] Fukatsu, Y., Shizuki, B. and Tanaka, J.: No-look flick: Single-handed and eyes-free japanese text input system on touch screens of mobile devices, *Proc. MobileHCI'13*, pp.161–170, ACM (2013).
- [8] 西田圭佑, 加藤恒夫, 山本誠一: スマートウォッチ向け日本語かな入力インターフェースの試作, インタラクシオン 2017, pp.897–902, 情報処理学会 (2017).
- [9] Leiva, L.A., Sahami, A., Catala, A., Henze, N. and Schmidt, A.: Text Entry on Tiny QWERTY Soft Keyboards, *Proc. CHI '15*, pp.669–678, ACM (2015).
- [10] 石井 晃, 箱田博之, 志築文太郎, 田中二郎: Flicky: 超小型タッチパネル端末におけるフリック操作を活用した QWERTY キーボード, 情報処理学会研究報告, Vol.2015-HCI-164, No.6, pp.1–8, 情報処理学会 (2015).
- [11] Shibata, T., Afegan, D., Kong, D., et al.: DriftBoard: A Pan-ning-Based Text Entry Technique for Ultra-Small Touchscreens, *Proc. UIST'16*, pp.575–582, ACM (2016).
- [12] Hong, J., Heo, S., Isokoshi, P. and Lee, G.: SplitBoard: A Simple Split Soft Keyboard for Wristwatch-sized Touch Screens, *Proc. CHI'15*, pp.1233–1236, ACM (2015).
- [13] Chen, X.A., Grossman, T. and Fitzmaurice, G.: Swipeboard: A Text Entry Technique for Ultra-small Interfaces That Supports Novice to Expert Transitions, *Proc. UIST'14*, pp.615–620, ACM (2014).
- [14] Darbar, R., Dash, P. and Samanta, D.: ETAO Keyboard: Text Input Technique on Smartwatch, *Proc. Procedia Computer Science*, Vol.84, pp.137–141 (2016).
- [15] 5-TILES, available from (<http://fivetiles.com>) (accessed 2019-05-28).
- [16] Komninos, A. and Dumlop, M.: Text Input on a Smart Watch, *Proc. IEEE Pervasive Computing*, Vol.13, No.4, pp.50–58 (2014).
- [17] Vertanen, K., Memmi, H., Emge, J., Reyat, S. and Kristensson, P.O.: VelociTap: Investigating Fast Mobile Text Entry Using Sentence-Based Decoding of Touchscreen Keyboard Input, *Proc. CHI'15*, pp.659–668, ACM (2015).
- [18] Mankoff, J. and Abowd, G.D.: Cirrin: A Word-Level Unistroke Keyboard for Pen Input, *Proc. UIST'98*, pp.213–214, ACM (1998).
- [19] Hinrichs, U., Schmidt, H., Isenberg, T., Hancock, M. and Carpendale, S.: BubbleType: Enabling Text Entry within a Walkup Tabletop Installation, Report 2008–893–06, *Department of Computer Science*, University of Calgary (2008).
- [20] Yeo, H.S., Lee, J., Bianchi, A. and Quigley, A.: WatchMI: Pressure touch, twist and pan gesture input on unmodified smartwatches, *Proc. MobileHCI'16*, pp.394–399, ACM (2016).

- [21] 齋藤航平, 奥 寛雅: HARI キーボード: 超小型タッチパネル端末向け日本語入力キーボード, インタラクシオン 2016, pp.701–703, 情報処理学会 (2016).
- [22] TouchOneKeyboard, available from (<http://www.touchone.net>) (accessed 2019-02-15).
- [23] 下岡純也, 山名早人: スマートウォッチにおけるタップ動作の少ない仮名文字入力手法, DEIM Forum 2017, I3-2, (2017).
- [24] 秋田光平, 田中敏光, 佐川雄二: スライドインによるスマートウォッチ向けの文字入力手法, インタラクシオン 2018, pp.276–281, 情報処理学会 (2018).
- [25] 梅澤 猛, 原 清貴, 大澤範高: 腕時計型端末における画面占有面積を抑えたかな文字入力手法, DICOMO 2017, pp.1746–1753, 情報処理学会 (2017).
- [26] 東條貴希, 加藤恒夫, 山本誠一: BubbleFlick: スマートウォッチ向け日本語かなフリック入力インターフェースの改良と 30 日間の連続評価, HCG シンポジウム 2017, 電子情報通信学会 (2017).
- [27] Tojo, T., Kato, T. and Yamamoto, S.: BubbleFlick: Investigating effective interface for Japanese text entry on smartwatches, *Proc. MobileHCI'18*, pp.44:1–44:12, ACM (2018).
- [28] Sheik-Nainar, M.: Contact location offset to improve small target selection on touchscreens, *Proc. CHI'15*, pp.1837–1846, ACM (2015).
- [29] Poulton, E.C. and Freeman, P.R.: Unwanted asymmetrical transfer effects with balanced experimental designs, *Proc. Psychological Bulletin*, Vol.66, No.1, pp.1–8 (1996).
- [30] Bangor, A., Kortum, P.T. and Miller, J.T.: An Empirical Evaluation of the System Usability Scale, *Proc. International Journal of Human-Computer Interaction*, Vol.24, pp.574–594 (2008).
- [31] 古井陽之助, 前田環貴, 松本誠義: 講義画面の自動連続キャプチャを行う講義・学習支援システムとその評価実験, 情報処理学会研究報告 (2014).
- [32] Run apps on the Android Emulator, available from (<https://developer.android.com/studio/run/emulator>) (accessed 2019-05-28).



加藤 恒夫 (正会員)

1994 年東京大学工学部電子工学科卒業。1996 年東京大学工学系大学院・電子工学専攻博士前期課程修了。同年国際電信電話株式会社入社。KDD 研究所, KDDI 研究所を経て, 2015 年より同志社大学理工学部インテリジェント情報工学科准教授。音声認識, ヒューマンインタフェースの研究開発に従事。2011 年情報処理学会喜安記念業績賞受賞。ヒューマンインタフェース学会, 日本音響学会, 言語処理学会, 電子情報通信学会, ACM, IEEE 各会員。博士 (情報理工学)。



山本 誠一 (正会員)

1950 年生。1972 年大阪大学工学部電子工学科卒業。1974 年同大学院基礎工学研究科修士課程修了。同年国際電信電話株式会社入社。ATR 音声言語コミュニケーション所長等を経て現在同志社大学理工学部教授。工博。この間, 適応信号処理, 音声合成, 音声認識, 音声翻訳, マルチモーダル対話の研究等に従事。1981 年度電子情報通信学会学術奨励賞, 日本音響学会第 3 回技術開発賞, 第 5 回技術開発賞, 電子情報通信学会システムソサイエティ論文賞, 電気通信普及財団テレコム技術賞等を受賞。日本音響学会, 言語処理学会, 人工知能学会各会員, IEEE Fellow, 電子情報通信学会 Fellow。



東條 貴希

2017 年同志社大学理工学部インテリジェント情報工学科卒業。2019 年同大学大学院理工学研究科情報工学専攻博士前期課程修了。同年サイボウズ株式会社入社。



本田 裕己 (正会員)

2019 年同志社大学理工学部インテリジェント情報工学科卒業。現在, 同大学大学院理工学研究科情報工学専攻博士前期課程に在籍。機械学習とクラウド技術に関心。