

QuadKey：キーの数を4つに限定したスマートウォッチ向け かな文字入力方式

安福 和史¹ 中村 喜宏^{1,a)}

受付日 2018年12月27日, 採録日 2019年5月9日

概要: 近年, スマートウォッチと呼ばれる腕時計型のウェアラブル端末が注目されている. スマートウォッチは既存の携帯型端末に比べて表示画面が格段に小さいため十分な大きさのキーを配置することができない. そのため, オクルージョンの問題やファットフィンガー問題が発生し, スマートウォッチで文字を入力することは困難である. キーボードとしての実用性を考慮すると, これらの問題の改善に加えて高い入力性能を兼ね備えている必要がある. そこで我々はタッチダウンキーから隣接キーへのスライドで入力可能な文字群を切り替えることにより入力に必要なキーの数を4つにしたかな文字入力方式である QuadKey を提案する. QuadKey はキーの数を4つにすることで各キーに十分な大きさを確保してオクルージョン・ファットフィンガー問題を改善する. さらに, 隣接キーへのスライドで文字群を切り替える仕組みにより, 1文字を1ストロークで入力することができ高い入力性能を有する. 評価実験の結果, 入力速度は79.7 CPM, エラー率は5.78%となりスマートウォッチのかな文字入力方式として有効性が示された.

キーワード: QuadKey, 文字入力, スマートウォッチ, ウェアラブル端末, ファットフィンガー, オクルージョン

QuadKey: A Japanese Kana Character Input Method for Smartwatches with Four Keys

TAKASHI YASUFUKU¹ YOSHIHIRO NAKAMURA^{1,a)}

Received: December 27, 2018, Accepted: May 9, 2019

Abstract: Recently, wearable devices called smartwatches have been attracting attention. Since a smartwatch has a much smaller display screen than existing mobile devices such as smartphones, it is not possible to arrange a full set of useable keys of sufficient size for it. Therefore, the occlusion problem and the fat finger problem occur, making it difficult to enter characters with smartwatches. Considering practicality of use as a keyboard, in addition to reducing these problems, it is necessary to have a high input performance. Thus, we propose a Japanese Kana character input method, QuadKey, that limits the number of keys to 4 by changing the character set assignment of a key by sliding to an adjacent key. Because the proposed method uses only 4 keys, a sufficient size is secured for each key and the occlusion and fat finger problems are reduced. Furthermore, the mechanism of changing the character set by sliding to the adjacent key allows one character to be input with a single stroke, realizing high input performance. Results of an experiment evaluate the proposed method as having an input speed of 79.7 CPM and an error rate of 5.78%, demonstrating its effectiveness as a Kana character input method.

Keywords: QuadKey, Japanese text entry, smartwatch, wearable device, fat finger, occlusion

1. はじめに

近年, スマートウォッチと呼ばれる腕時計型のウェアラブル端末が注目されている. スマートウォッチは, 主に電話やメール, SNS の通知, 各種センサを用いた健康管理やナ

¹ 日本大学大学院生産工学研究科
Graduate School of Industrial Technology, Nihon University,
Narashino, Chiba 275–8575, Japan

a) nakamura.yoshihiro@nihon-u.ac.jp

ビゲーションに利用されている。音声入力や QWERTY・フリックキーボードなどのソフトウェアキーボードを備えており簡単な文章を入力することも可能である。しかし、音声入力は雑音環境下で認識率が低く、プライバシーやセキュリティの観点から使用できる環境に限られる。現状のソフトウェアキーボードはスマートウォッチの小さい領域に 12 から 30 個程度のキーを配置する必要があり各キーに十分な大きさを確保することができない。そのため、キーが指に隠れてしまい入力の妨げとなるオクルージョンの問題、キーが指よりも小さくなり押下したいキーを正確に押下できないファットフィンガー問題 [1], [2] が発生し、スマートウォッチで文字を入力することは非常に困難である。これらの問題を改善した小型端末向けのキーボードは、いくつか提案されているが、その多くは QWERTY キーボード (アルファベット入力) をベースとした方式であるため、かな文字を入力するには複数回の操作が必要となり入力時間とエラー (誤入力) の増加につながる。また、スマートウォッチの文字入力にはオクルージョン・ファットフィンガー問題の改善は必須だが、これらを改善するだけで入力性能がともなわなければ実用的とはいえない。

そこで我々は、キーの数を 2×2 の 4 つに限定したフリックキーボードベースのスマートウォッチ向けかな文字入力方式である「QuadKey」を提案する。QuadKey は各キーに 3 つの文字群 (合計 15 文字) が割り当てられており、タッチダウンキー上でのフリックで 1 文字群 (5 文字) を入力し、タッチダウンキーから水平方向もしくは垂直方向に隣接しているキーへ指をスライドすることでスライド方向に応じて残り 2 つの文字群に切り替えて入力を行う。この仕組みにより 4 キーで 60 文字の入力が可能となり、またフリックによる入力とスライドによる文字群の切替えを 1 ストロークでシームレスに行うことができる。

QuadKey はキーの数を 4 つに限定し各キーに十分な大きさを確保することでオクルージョン・ファットフィンガー問題の改善を図り、かつ文字をパターンとして 1 ストロークで入力できることから高速な文字入力およびエラーの少ない入力が可能であると考えられる。QuadKey のスマートウォッチにおけるかな文字入力方式としての有効性を検証するために評価実験を行った。

2. 関連研究

スマートウォッチのための文字入力については様々な方式が提案されている。

2.1 手書き文字認識方式

手書き文字入力方式はディスプレイ上に 1 文字ずつ手書き入力する方式である [3], [4]。キーボードと比較して入力速度が遅いのと、文字の区切りの判断が難しいという問題がある。この問題を改善する方法として一筆書き入力があ

る。EdgeWrite [5] では矩形の入力領域の 4 つの頂点を結ぶ一筆書きでアルファベットを入力する。同じくカタカナ入力するものも開発されている [6]。しかし、これらの方式は特殊な一筆書き文字を覚える必要があり学習の負荷が高いという問題がある。

2.2 ソフトウェアキーボード

ソフトウェアキーボードの場合はキーの大きさが小さくなるため指でキーを押し分けるのが難しくなるファットフィンガー問題やキーが指で隠れて見えなくなるオクルージョンの問題が発生する。これを改善するためにキーの数を減らすことが有効である。

2.2.1 アルファベット入力方式 (独自)

FlickKey Mini Keyboard [7] では 8 方向のフリック入力により 1 つのキーに 9 文字割り当てることでキーの数を削減している。そのため、より正確なフリック操作が要求される。5-TILES [8] は 5 から 6 個のアルファベットが割り当てられた 5 つのキーを横一列に配置しており、キーにタップするかタッチした指をスライドさせ目的のキーの上で離すことにより文字を入力する方式である。画面の横幅に 5 個のキーを配置するためキーの大きさが小さくなるという問題点がある [9]。

2.2.2 アルファベット入力方式 (QWERTY ベース)

QWERTY 方式は物理キーボードの使用によりユーザがキー配置に慣れているためスマートフォンなどでも多く利用されている。しかし、最低でも 30 個以上のキーが必要なため、スマートウォッチではキーの大きさが小さくなりすぎファットフィンガー問題やオクルージョンの問題が発生する。この問題の緩和を目指していくつかの研究が行われている。ZoomBoard [10] はタップによりキーボードを段階的に拡大表示することで、表示されるキーの数を減らして選択しやすくする方法である。キーボードそのものを拡大するのではなく指で隠された部分をふきだし部分に拡大表示し、指の動きで選択する ZShift [11] やフリック操作で選択する Flickey [12] などがある。SplitBoard [13] は左右 2 分割されたキーボードをスワイプにより選択した後、所望のキーをタップする方法である。Swipeboard [14] は QWERTY キーボードを 9 つに分割した領域をスワイプにより選択した後、領域に含まれる 3 文字から所望の文字をスワイプで選択する方式である。上記の QWERTY 方式をベースとした方式はキーボードを拡大表示するか段階的な選択によりキーを絞り込むことでキーの数の削減を図っている。

2.2.3 辞書を用いた予測入力方式

別のアプローチとしてキーに複数の文字を割り当ててキーストロークから統計的な辞書を使って英単語を予測する方式がある。TouchOne [15] はアルファベット順に 3 から 4 文字を 1 つのキーとして 8 つのキーを画面の外周

に沿って配置した方式である。MinimumKeyboard [16] はキーを横一列に配置した方式である。これによりキーの数を減らすことができるが予測には曖昧性があるため複数の候補から所望の単語を選択する必要がある。また辞書にない単語は1文字ずつ直接入力する必要がある。

2.2.4 かな文字入力方式

かな文字は文字数がアルファベットの2倍近く多いため画面が極端に小さいスマートウォッチで快適な入力を実現するのはアルファベットより難しい。そのため、子音（行）と母音（段）を指定することで入力する方式が多く提案されている。フリック方式はキー選択で子音（行）を選択し、タップと4方向のフリックで母音（段）を選択することにより文字を入力する方式である。1ストロークで1文字入力できるため高速入力が可能である。しかし、スマートウォッチに適用するとキーの数が12個と多くキーが小さくなり押し分けにくくなるという問題点がある。HARIキーボード [17] は円形の画面の周囲に子音が配置されており子音にタッチした指を画面の中心まで動かすと画面の周囲に母音が配置され、そのまま指を入力したい母音の上まで動かしタッチアップすることで1ストロークで文字が入力できる方式である。BubbleFlick [18] は円形の画面の周囲に各行のあ段の文字を環状に配置し文字にタッチするとその周りに放射状に当該行の文字が表示され、対象文字の方向にフリックすることで文字を入力する方式である。さらに、この方法を改良したBubbleSlide [19] では環状に配置した1段目の文字をタッチすると画面中心に向かって当該行の文字が1列に並ぶように表示し、指をスライドして選択する方式である。尾崎らの方式 [9] は角型の画面を対象とし、行は画面の四隅のいずれかから始まるストロークジェスチャで指定し、段はその後表示される当該行の文字をタップで指定する方式である。主にキーボードの画面の占有率低さを目的としている。ShuttleBoard [20] は10個の子音キーと5個の母音キーから構成され子音キーから母音キーに指でなぞることで1文字が入力できる方式である。この動作を指を離さずに行うことで複数の文字を連続して入力することもできる。キーの数が15個と多いためキーの大きさが小さく、そのためエラー率が10%を超えている。No-look Flick [21] はスマートフォン向けのアイズフリーな文字入力方式である。画面を3つの領域に分割し、そのうちの2つの領域を子音に、1つを母音に割り当てて、子音、母音の順に2回のフリックで1文字を入力する。この方式はキーが3個あれば入力できるためそのままスマートウォッチへの適用も可能と考えられる。

本研究では、かな文字を1ストロークで入力可能なフリック方式の利点を生かしながら、ファットフィンガーやオクルージョンの問題を改善し、さらに入力性能に優れたスマートウォッチ向けかな文字入力方式について検討する。

3. QuadKey

QuadKey は 2×2 の4つのキーで構成されたフリックキーボードベースのスマートウォッチ向けかな文字入力ソフトウェアキーボードである（図1）。各キーに自キー上で入力できる1つの文字群（以降、メイン文字群）と、隣接キー中央にスライドすることで入力できる2つの文字群（以降、サブ文字群）の合計3つの文字群が割り当てられている。キーに対してタッチダウンすることでメイン文字群の入力状態となるが、水平方向もしくは垂直方向に隣接しているキー中央へ指をスライドすることで方向に応じたサブ文字群の入力状態へと切り替える仕組みを備えている（図2）。この仕組みにより、4つのキーのみでも60文字（5文字 \times 3方向 \times 4キー）を割り当てることができ、「かな文字46個」と「特殊文字・機能14個」を入力することが可能である。

QuadKey はキーの数を4つに限定することで1つのキーサイズを大幅に大きくすることができ、さらに 2×2 のキー配置のため正確にキーの位置を把握する必要がなくキーが指に隠れてしまっても入力の大きな妨げにならない。また、清音（濁点や小文字などの特殊文字を除いた文字）を1つのパターンとして記憶して、1ストロークで入力することができるため習熟すれば高速かつエラーの少ない文字入力が期待できる。

以上から、QuadKey はオクルージョン・ファットフィンガー問題を改善し、かつ高い入力性能を有していると予想されスマートウォッチのかな文字入力方式として有効であると考えられる。

3.1 入力方法

文字の入力例を図3に示す。図3は「こんにちは」と入力した場合の例である。図3から分かるように、QuadKey



図1 QuadKey
Fig. 1 QuadKey.

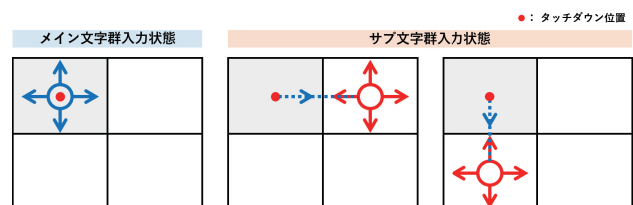


図2 QuadKey の入力方式
Fig. 2 QuadKey input method.

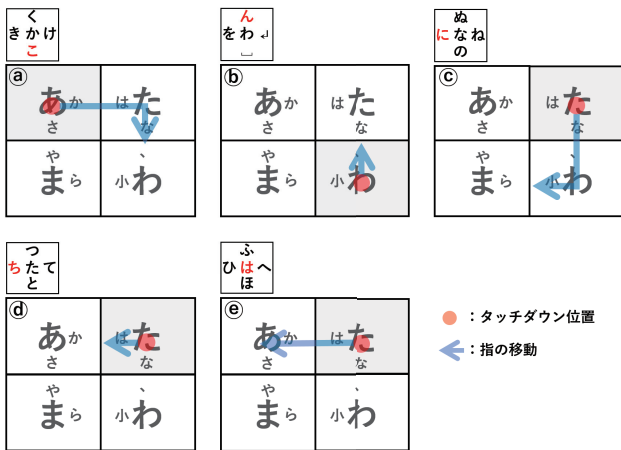


図 3 文字入力例 (入力文字: こんにちは)
Fig. 3 Examples of text entry.

は自キー上での入力 (メイン文字群の入力) とスライド先キー上での入力 (サブ文字群の入力) に分かれています。キーの選択, メイン文字群の入力, サブ文字群の入力, 特殊文字・機能の入力 (選択) について以下に説明する。

3.1.1 キーの選択

各キーの中央にはメイン文字群, その周囲の隣接キー側には対応した方向のサブ文字群の「あ段」の文字が表示されている (以降, ガイド)。ガイドに従い入力したい文字が割り当てられているキーに対してタッチダウンする。キーにタッチダウンすることで, そのキーに割り当てられているメイン文字群の入力状態が開始される。

3.1.2 メイン文字群の入力

入力したい文字がメイン文字群の場合は, タッチダウンしたキー上でタッチアップ・フリック操作を行う。タッチアップ (タップ) で「あ段」, 左フリックで「い段」, 上フリックで「う段」, 右フリックで「え段」, 下フリックで「お段」の文字が入力される。

3.1.3 サブ文字群の入力

入力したい文字がサブ文字群の場合は, タッチダウンしたキーからガイドに従い水平方向もしくは垂直方向に隣接しているキーの中央に指をスライドしてサブ文字群の入力状態に切り替える。そして, スライドしたキー上でタッチアップ・フリック操作を行う。タッチアップで「あ段」, 左フリックで「い段」, 上フリックで「う段」, 右フリックで「え段」, 下フリックで「お段」の文字が入力される。なお, 一度サブ文字群の入力状態に切り替えた場合はメイン文字群の入力状態に戻せない仕様となっている。これは文字群の修正よりもサブ文字群に切り替わった後に大きくフリックできるメリットを優先したからである。

3.1.4 特殊文字・機能の入力 (選択)

「改行/確定」や「空白/変換」などの特殊文字・機能も, 通常の文字と同様にメイン文字群もしくはサブ文字群として割り当てられているため, 同様の方法で入力 (選択) す

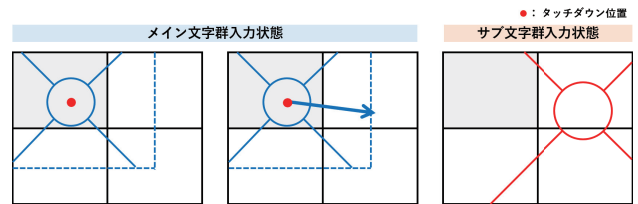


図 4 文字判定アルゴリズム
Fig. 4 Character decision algorithm.

ることが可能である (3.3 節参照)。たとえば, 「空白/変換」は右下キーのメイン文字群である下フリックに割り当てられており, 文字の確定前入力中に右下キーを下フリックすることで漢字・カタカナに変換したり, 変換候補の選択が可能である。

以上のように, QuadKey はフリックとフリックを延長したスライドを用いることで「文字群 (子音, 行) の選択」と「文字 (母音, 段) の選択」を 1 ストロークでシームレスに行うことが可能である。

3.2 文字判定アルゴリズム

前節の「入力方法」では分かりやすさを優先して「タッチアップ」や「上フリック」など大雑把な説明を行ったため本節ではより詳細な文字判定のアルゴリズムを説明する。左上キーにタッチダウンした場合のメイン文字群の入力状態と, そこから右上キーにスライドしたサブ文字群の入力状態を図 4 に示す。各入力状態の文字判定を以下に説明する。

3.2.1 メイン文字群の入力状態

タッチダウン位置を中心に直径がキー横幅の 50% の円形領域 (あ段入力領域) と, その周囲からタッチダウンキーの 1.5 倍の範囲, つまり隣接キー中央まで X 型に上下左右のフリック領域 (左: い段入力領域, 上: う段入力領域, 右: え段入力領域, 下: お段入力領域) を展開する (図 4 左)。展開された円形領域でタッチアップすることで, メイン文字群の「あ段」の文字が入力され, 上下左右のフリック領域のいずれかでタッチアップすることで, メイン文字群のタッチアップ位置に「対応した段」の文字が入力される。

円形領域を設けたのは, タップ動作によって「あ段」の文字を入力する際に, タッチダウン位置とタッチアップ位置のずれによってタップ動作がフリック動作だと誤認識されてしまうのを軽減するためである。フリック領域をキーサイズの 1.5 倍までにするのは, スライドではない小さい指の移動 (フリックなど) による入力状態の誤切替えを軽減するためである。また, キーの外側付近にタッチダウンした場合は, キーボードをはみ出した部分が欠けた状態で円形領域が展開され, 欠けた方向のフリック領域は十分に展開されないが, キーボードの周囲には数 mm 程度の余白部分があるため, そちら側にフリックすることで欠けた方向側に割り当てられている文字を入力することができる。つ

まり、欠けた方向の余白部分が、展開されなかったフリック領域として機能する。たとえば、左上キーの左端付近にタッチダウンした場合は、キーボードをはみ出した左部分が欠けた状態で円形領域が展開され、左フリック領域は十分に展開されないが、左側の余白部分にフリックすることで左フリック領域に割り当てられている文字を入力することができる。

図4中央のように、展開されたフリック領域外、つまり隣接キー中央までスライドするとサブ文字群の入力状態に切り替わる。

3.2.2 サブ文字群の入力状態

スライドによってサブ文字群の入力状態に切り替わった位置を中心に直径がキー横幅の60%の円形領域と、その周囲からキーボード全体までX型に上下左右のフリック領域を展開する(図4右)。展開された円形領域でタッチアップすることで、サブ文字群の「あ段」の文字が入力され、上下左右のフリック領域のいずれかでタッチアップすることで、サブ文字群のタッチアップ位置に「対応した段」の文字が入力される。

円形領域をメイン文字群の入力状態より大きくしたのは、円形領域でタッチアップしようとしているにもかかわらず、隣接キーからのスライドの慣性などにより、その先にあるフリック領域でタッチアップしてしまう誤入力を低減するためである。また、キーの外側付近にスライドした場合は、キーボードをはみ出した部分が欠けた状態で円形領域が展開され、欠けた方向のフリック領域が十分に展開されないが、メイン文字群の場合と同様に、欠けた方向の余白部分が、展開されなかったフリック領域として機能するため、欠けた方向に割り当てられている文字を入力することができる。

3.3 文字の割当て

各キーには1つのメイン文字群と2つのサブ文字群が割り当てられている。タッチ位置でのタッチアップ(タップ)に「あ段」、左フリックに「い段」、上フリックに「う段」、右フリックに「え段」、下フリックに「お段」が割り当てられており、入力状態の文字群の対応した方向(位置)の文字が入力される。

各キーに割り当てられている文字を表1に示す。上から左上キー、右上キー、左下キー、右下キーに割り当てられている3つの文字群を示している。各キーの一番上の1行がメイン文字群、残りの2行がサブ文字群の割当てである。スライド先列の「→キー名」はそのキー名へのスライドを表している。左上キーを例として説明する。メイン文字群には「あ行」が割り当てられている。右上キーへスライドした場合のサブ文字群には「か行」が割り当てられている。左下キーへスライドした場合のサブ文字群には「さ行」が割り当てられている。

表1 文字の割当て

Table 1 Assignment of characters.

タッチ	スライド先	中央	左	上	右	下
左上	→右上 →左下	あ	い	う	え	お
		か	き	く	け	こ
		さ	し	す	せ	そ
右上	→右下 →左上	た	ち	つ	て	と
		な	に	ぬ	ね	の
		は	ひ	ふ	へ	ほ
左下	→左上 →右下	ま	み	む	め	も
		や	ゆ	よ	ゆ	よ
		ら	り	る	れ	ろ
右下	→左下 →右上	わ	を	ん	[改/確]	[空/変]
		[記号]	[BS]	[英]	[数]	[小]
		,	.	?	!	-

角括弧：特殊文字・機能(改行/確定, 空白/変換, 記号モードなど)

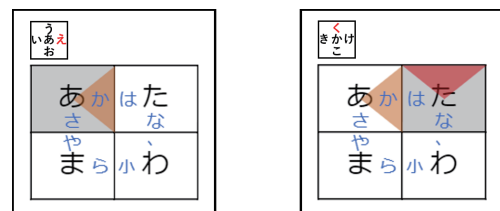


図5 視覚フィードバック

Fig. 5 Visual feedback.

日本語の出現頻度を考慮した効率的な文字の割当てがあるかもしれないが、本研究では習熟のしやすさを優先し、五十音を左上キー、右上キー、左下キー、右下キーの順に先頭から3行ずつ割り当てた。また、各キーに2つずつあるスライド先へのサブ文字群の割当てでは、タッチダウンキーを中心とした上下左右の4方向において、2つのスライド先のある方向が時計回りに連続するようになる順番で、五十音の若いサブ文字群を割り当てた。

3.4 視覚フィードバック

QuadKeyはユーザの入力を補助するための視覚的なフィードバックを提供する(図5)。キーにタッチしている間、キーボード上部に入力状態の文字群をフリック方向に対応した配置で表示する(現在選択されている文字は赤字)。また、タッチ状態のキーに対して指を動かしている方向側に開いたオレンジ色の三角形を表示する。サブ文字群入力状態になった場合はスライド元キーの三角形を残しつつ、スライド先キーで同様に指を動かしている方向側に開いた赤色の三角形を表示する。これらの視覚フィードバックは入力状態の文字群と文字、フリック方向、スライド方向、スライド元キーを確認することができ文字の割当てをまだ覚えていない初心者の入力支援と文字の割当ての学習支援を目的としている。

4. 評価実験

4キー(2×2)のキーボードがどの程度、オクルージョン・ファットフィンガー問題に対して有効かを確かめるために同面積内で2×2, 3×3, 4×4, 5×5の配列のキーボードを用意してキーのタップ実験を行った。続いて, QuadKeyの入力速度とエラー率を検証するためにNo-look Flick [21]を比較対象として, 想定したスマートウォッチ上部外側に表示される単語を入力していく文字入力実験を行った。実験は大学生と大学院生の男性7名(年齢:21~26歳)に協力してもらった。実験に参加した7名は全員が右利きであり, ふだんからスマートフォンでフリックキーボードを用いた文字入力を行っていた(利用歴:8~10年)。また, それぞれの実験ではスマートウォッチを想定した実験用プロトタイプキーボードをスマートフォン上に実装し, それを腕にリストバンドで装着して実験を行った。使用したスマートフォンは, Android 端末のZenFone2 Laser(画面サイズ:5インチ, 解像度:1,280×720)である。

4.1 タップ実験

4.1.1 用意したキーボード

想定したスマートウォッチの画面サイズは, 関連研究の多くで用いられている24mm×24mmとした。24mm×24mmのうち, 入力フォームやフリックのための余白を考慮して, 20mm×16mmをキーボード領域とした。この20mm×16mmちょうどに収まるように, 以下の4つの配列のキーボードを用意した。括弧内は, 1キーのサイズを示している。

- ① 2×2 (10.00mm×8.00mm)
- ② 3×3 (6.66mm×5.33mm)
- ③ 4×4 (5.00mm×4.00mm)
- ④ 5×5 (4.00mm×3.16mm)

各キーボードには左上のキーから順に, 「1~キー数」までの数字が表示されている。

4.1.2 実験手順

実験協力者7名にタップ実験について説明を行った。実験順序が与える影響を打ち消すために, 各実験協力者のキーボードの実験順序はランダムに割り振りを決定した。実験協力者には, 「1~キー数」の範囲でランダムに表示される数字のキーをタップしてもらった。25タップを1セッションとして, セッションごとに3分程度のインターバルを置いて合計4セッションの実験を行った。4セッションが終了次第, 同様の手順で次の配列のキーボードの実験を順次行った。実験は大学内の静かな実験環境で, インターバル時間などを管理する監督者の立ち会いのもと行った。

なお, タップ実験において, タップされたキーは正誤にかかわらず, 次の数字に進むものとした。

4.2 文字入力実験

4.2.1 比較対象

文字入力実験におけるQuadKeyの比較対象としてNo-look Flickを選択した。No-look Flickは2つの子音キーと1つの母音キーで構成されており, フリックキーボードをベースとしたスマートフォン向けのアイズフリーかな文字入力方式である。基本的には子音キーで子音を選択し, 続いて母音キーで母音を選択して1文字の入力を行う。スマートフォン向けに開発された方式だが, 3つのキーのみでかな文字を入力できることからスマートウォッチのかな文字入力としても非常に有効であると考えた。No-look Flickは3キー(母音キーが2キー分のサイズのためキーボードサイズは実質4キー)を用いて2フリックで1文字を入力する方式であり, QuadKeyは4キーを用いて1ストロークで1文字を入力する方式である。よって, 使用するキーの数がほぼ同じ方式として, 1ストローク入力と2フリック入力のどちらがスマートウォッチの入力に適しているか比較検証できることからNo-look Flickを比較対象として選択した。

4.2.2 用意したキーボード

想定したスマートウォッチの画面サイズは, タップ実験と同様に, 24mm×24mmとし入力フォームやフリックのための余白を考慮して, 20mm×16mmをキーボード領域とした。そして, 20mm×16mmちょうどに収まるようにQuadKeyとNo-look Flickを実装した(図6)。キーボード左上部には視覚フィードバックの領域, キーボード右上部には6文字分の入力フォームの領域を設定した。なお, 本来のNo-look Flickはスマートフォンのタッチ位置の分布を考慮して, 母音キーよりも子音キーの横幅を大きくしているが, 今回は想定するキーボードサイズが小さいため同じ横幅とした。

4.2.3 用意した単語

国立国語研究所の現代日本語書き言葉均衡コーパス(BC-CWJ)の頻度リスト [22]から抽出した450単語を用いた。抽出条件として, 数詞と助数詞は「二十一(21)」「二十二(22)」の「にじゅう」, 「一枚」「二枚」の「まい」のように単語に含まれる文字の偏りが大きいため除外し, 読みが同一の単語は1つに統合した。そして, 頻度順にひらがなで

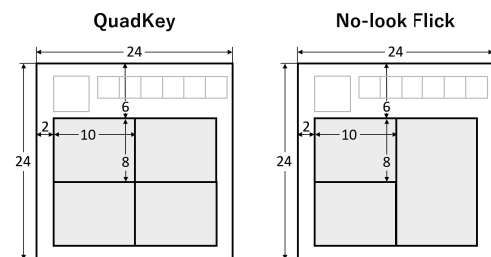


図6 実験用のキーボードのサイズ (mm)

Fig. 6 Size of prototype keyboards.

4文字, 5文字, 6文字になる単語をそれぞれ150個ずつ, 合計450単語を抽出した。

本実験で使用する文字は, かな50音, 濁点(°), 半濁点(ˆ), 小文字, 長音符(ー)のみである。

4.2.4 評価指標

文字入力実験では入力速度とエラー率を測定した。文字入力速度の評価指標として1分間あたりの文字入力数であるCPM (Characters Per Minute)を用いた。CPMは値が大きいほど良好な結果といえる。エラー率の評価指標として修正済みエラー率であるCerr (Corrected Error Rate) [23]を用いた。Cerrは文字入力実験において修正した文字数を全入力文字数で割り100を掛けた値である。Cerrは値が小さいほど良好な結果といえる。

4.2.5 実験手順

実験協力者7名にQuadKeyとNo-look Flickの入力方法を十分に説明した。次に, 各キーボードに1分程度の練習時間を設け, 入力方法に疑問がないことを確認した。実験順序が与える影響を打ち消すために, 各実験協力者の2つのキーボードの実験順序はランダムに割り振りを決定した。実験協力者には, 想定したスマートウォッチ領域の上部外側に表示される単語を入力してもらった。450単語の中から重複なくランダムで抽出した5単語(平均25文字)を1セッションとして, セッションごとに3分程度のインターバルを置いて1日に合計5セッションの実験を行った。文字入力実験では, ある程度の習熟をしてもらうために上記の実験を30日間実施して, 最終的には合計150セッション(750単語, 平均3,750文字)の実験を行った。セッション間で重複して出現する単語もあるため, 平均して1単語あたり1.67回入力したことになる。150セッションが終了次第, 同様の手順で次のキーボードの実験を行った。実験は大学内の静かな実験環境で, インターバル時間などを管理する監督者の立ち会いのもと行った。

文字入力実験において, 入力された文字は1文字ごとに正誤を判定して正解の場合のみ次の文字の入力に進み, 単語の最後の文字まで正解した場合は自動で次の単語が表示されるものとした。

5. 評価結果

5.1 タップ実験

各配列キーボードのエラータップ率(全セッションのエラータップ数を全セッションの合計タップ数で割って100を掛けた値)を図7に示す。エラータップ率は2×2のキーボードが0.3%, 5×5のキーボードでは21.1%と必然的な結果だが, キーの数が多くなりキーサイズが小さくなるにつれて大幅に上昇している。また, 各キーボードのエラータップ原因を分類したところ, そのすべてが正解キー周囲のキーへのタップであった。

タップ実験は表示された数字のキーをタップするだけの

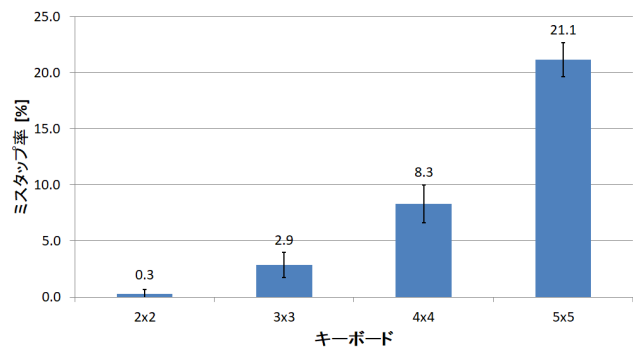


図7 各配列のエラータップ率

Fig. 7 Error tap rate of each key arrangement.

表2 各実験協力者における最終日の結果

Table 2 Results on the final day of each experimenter.

	QuadKey		No-look Flick	
	CPM	Cerr [%]	CPM	Cerr [%]
#1	86.7 (4.5)	4.59 (3.7)	67.6 (2.5)	4.43 (3.9)
#2	76.6 (2.8)	6.50 (3.6)	63.4 (2.0)	5.08 (3.2)
#3	78.3 (2.1)	6.17 (3.7)	64.8 (2.8)	6.82 (5.8)
#4	84.1 (3.7)	4.93 (2.8)	62.2 (1.8)	5.49 (2.5)
#5	75.8 (2.8)	4.63 (1.5)	66.7 (3.1)	6.70 (2.6)
#6	77.2 (3.5)	8.21 (5.2)	65.3 (2.9)	4.15 (2.8)
#7	79.5 (1.7)	5.47 (3.7)	61.7 (3.4)	6.06 (2.3)
平均	79.7 (4.0)	5.78 (1.3)	64.5 (1.8)	5.53 (1.1)

シンプルで複雑な操作を必要としない実験である。よって, エラーの原因をすべて出題数字およびキー位置の誤認識によるものではないと仮定するならば, この実験におけるエラータップ率の上昇は純粋なオクルージョン・ファットフィンガー問題の影響によるものであると考えられる。つまり, 単純に入力したいキーに対してタッチする場合において, 2×2のキーボードはオクルージョン・ファットフィンガー問題の改善に有効であり, それはQuadKeyでも同様であるといえる。

5.2 文字入力実験

5.2.1 入力性能

QuadKeyとNo-look Flickの各実験協力者における最終日(30日目, 146~150セッション)の平均入力速度および平均Cerrを表2に示す。括弧内の数字は標準偏差を示している。

最終日における全実験協力者の平均入力速度はQuadKeyが79.7CPM, No-look Flickが64.5CPMとなり, QuadKeyのほうが15.2CPMほど速い結果となり, 平均CerrはQuadKeyが5.78%, No-look Flickが5.53%となり, QuadKeyのほうが0.25%ほど高い結果となった。QuadKeyは標準偏差が大きいことからNo-look Flickに比べて実験者ごとに結果のばらつきが大きく, 習熟に個人差が出やすいと判断できる。また, 30日目時点での休憩時間を含まない

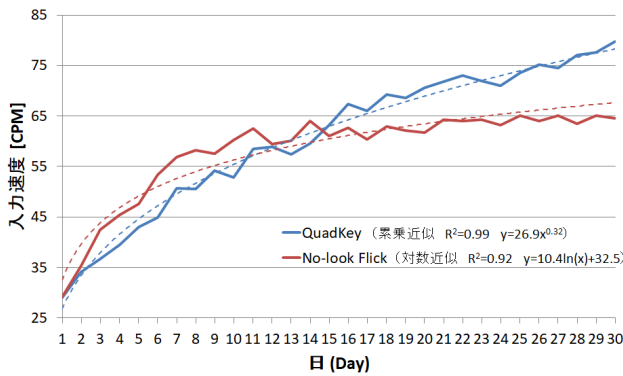


図 8 各日における平均入力速度

Fig. 8 Average text entry speed (CPM) for each day.

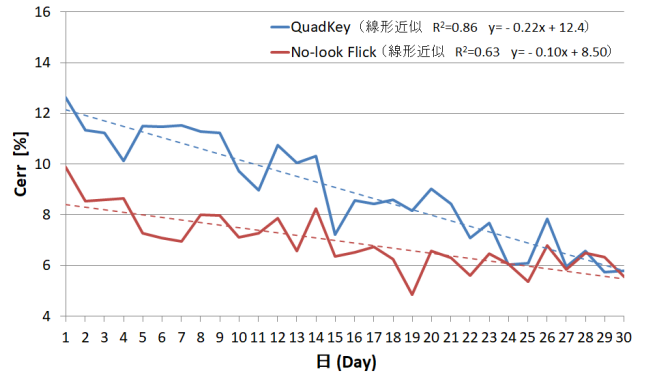


図 9 各日における平均 Cerr

Fig. 9 Average Cerr for each day.

平均総入力時間は約 66 分である。

QuadKey と No-look Flick の最終日の入力速度および Cerr に有意差があるか t 検定を行った。入力速度において有意水準 5% で片側 t 検定を行ったところ、 $t(6) = 8.95$, $p = 0.00 < 0.05$ で有意差が認められた。Cerr において有意水準 5% で両側 t 検定を行ったところ、 $t(6) = 0.34$, $p = 0.75 > 0.05$ で有意差は認められなかった。

5.2.2 入力速度の変化

QuadKey と No-look Flick の 1 日 (5 セッション) ごとの全実験協力者の平均入力速度を図 8 に示す。図中の破線は QuadKey と No-look Flick の入力速度をそれぞれ累乗近似, 対数近似したものである。近似曲線の種類は決定係数 (R^2 値) が最も 1 に近いものを選択した。

1 日目, 2 日目は各キーボードともに同等の入力速度であるが, 3 日目からは徐々に差が広がっていき 10 日目あたりまで 5~8 CPM ほど QuadKey のほうが遅い。各キーボードの近似曲線関数を微分して求めた 1 日目の傾きは QuadKey が約 2.9, No-look Flick が約 4.2 であることから出だしにおける入力速度の上昇率に差があることが分かる。これは QuadKey は No-look Flick と比較して入力方法が複雑なため習熟が難しいからだと考えられる。しかし, 10 日目以降から No-look Flick の上昇がきわめて緩やかとなり, 15 日目には QuadKey が追い抜いて徐々に差が広がっていき, 30 日目には約 15 CPM ほど QuadKey のほうが速くなっている。30 日目時点の No-look Flick の傾きは約 0.05 でグラフからも習熟傾向にあると判断できるが, QuadKey の傾きは約 0.15 と No-look Flick の 3 倍あり, グラフからも依然上昇傾向にあると考えられる。

全体的に QuadKey は緩やかに入力速度が上昇しており 30 日目の時点でも上昇傾向にあることから習熟に時間がかかることが分かる。

5.2.3 エラー率の変化

QuadKey と No-look Flick の 1 日 (5 セッション) ごとの全実験協力者の平均 Cerr を図 9 に示す。図中の破線は各キーボードの Cerr を線形近似したものである。近似曲

表 3 既存方式の入力速度とエラー率

Table 3 Input speeds and error rates of existing methods.

方式名	(字種)	CPM	Cerr[%]	備考	出典
フリック	(かな)	90.7	11 [~]		文献 [20]
ZoomBoard	(英字)	22.9	5.9	large	文献 [11]
		24.9	2.01	large	文献 [12]
		34.7	7 [~]		文献 [20]
ZShift	(英字)	25.4	11.4	large	文献 [11]
		34.6	6.52	large	文献 [12]
Flickey	(英字)	24.6	9.48	large	文献 [12]
Swipeboard	(英字)	54.7	16.9 [~]		文献 [14]
		30.5	6 [~]		文献 [20]
ShuttleBoard	(かな)	49.2	12 [~]		文献 [20]
尾崎らの方式	(かな)	41.8	3.86*	熟練者	文献 [9]
BubbleSlide	(かな)	44.4	10*	全員	文献 [19]
No-look Flick	(かな)	64.5	5.53		本稿
QuadKey	(かな)	79.7	5.78		本稿

[~]: グラフより読み取った値, 計算した値 * : Cerr 以外の評価指標
 ※ 文献により実験期間や事前練習の内容, 1 セッションあたりの入力文字数, 用いた単語などの実験条件が異なる点に留意

線の種類は決定係数 (R^2 値) が最も 1 に近いものを選択した。

1 日目の平均 Cerr は QuadKey が約 12.6%, No-look Flick が 9.87% と約 2.7 ポイントほど QuadKey が高い。これは, 前項の「入力速度の変化」でも述べたように, QuadKey の入力方法が No-look Flick よりも複雑なため慣れていない状態だとエラーが多くなるからだと考えられる。また, 各キーボードの線形近似曲線関数の傾きは QuadKey が約 -0.22 , No-look Flick が約 -0.10 であり, 1 日ごとに 0.12 ポイントずつ差が縮まっていく。30 日目には QuadKey が 5.78%, No-look Flick が 5.53% となり, ほぼ同等のエラー率となった。

6. 考察

参考文献より調査した既存方式の入力速度とエラー率を表 3 に示す。英字方式の CPM はローマ字変換で入力したとして文献内の WPM から, かな文字の CPM に換算し

たものである (WPM \times 5/1.79). このとき文献 [9] を参考にアルファベット 1 文字を入力するのに必要な文字数を 1.79 文字と仮定した. 複数の条件で実験されているものは本研究の実験条件に最も近いものを選択し, 表 3 の備考欄に選択した条件を記載した. また, いくつかの文献で実験されていた方式は複数の結果を示している. 基本的には, 最終結果として文献内に示されていた入力速度およびエラー率を引用し, 示されていないものはグラフから読み取った値, もしくは計算した値を引用した. 文献により実験期間や事前練習の内容, 1 セッションあたりの入力文字数, 用いた単語などの条件が異なるため正確な比較はできないが大まかな目安として考察の参考とする.

6.1 オクルージョン・ファットフィンガー問題

QuadKey はオクルージョン・ファットフィンガー問題を改善する方法としてキーの数を 4 つに限定することで 1 つ 1 つのキーサイズを大きくするというアプローチをとった. タップ実験の結果, 4 キー (2 \times 2) のキーボードのミスタップ率は 0.3% と非常に小さい値となった. さらに, QuadKey の 30 日目時点における Cerr の原因を分類したところエラーの多くはスライド方向の間違いであり, タッチダウンキーの間違いは 5.78% のうち, わずか 0.5% とタップ実験の結果とほぼ同等であった. エラーの大部分であるスライド方向の間違いは習熟不足に起因するものと考えられ, 所望のキーに対しては正確にタッチダウンできている. このことから, キーの数を 2 \times 2 の 4 つにした QuadKey はキーの押し分けが容易になっており, また高い入力速度がでていることからキーの数を減らしてキーサイズを大きくするというアプローチがオクルージョン・ファットフィンガー問題をある程度改善できていると推察できる.

6.2 入力性能

表 3 より, QuadKey は他方式と比較しても高い入力性能を有していることが分かる. 特に, 入力速度はフリック方式について高く, エラー率もその他の方式と比較しても遜色のない値であるといえる. 入力速度ではフリック方式に及ばなかったものの, エラー率が半分程と低く, ミスタイプが少ないことからスマートウォッチのような小型端末の文字入力としては QuadKey の方がストレスなく快適に入力ができると考えられる. しかし, 先にも述べたように文献によって実験方法などの違いや, それぞれが目的に応じた独自の特徴を持っており, 一長一短があるため, 表 3 の数値のみでどちらが良いかを単純に論じることはできない.

QuadKey の高い入力性能の理由について以下に考察する. QuadKey は 1 ストロークによる入力, 入力の起点となるキーが 2 \times 2 の 4 つであること, 母音選択のフリック

方向が共通であることが高い入力性能を発揮できた要因であると考えられる. 習熟により文字ごとのストローク操作をパターンとして覚えることができ, さらに 1 文字の入力に要するキーへのタッチダウン・タッチアップ操作が 1 回で済むため高速な文字入力が可能となる. 元々はオクルージョン・ファットフィンガー問題の改善を目的としたものだが入力の起点となるキーを 2 \times 2 の 4 つにしたことでキーの位置を把握しやすくなり, 所望のキーのターゲティングに要する時間の削減にもつながったと考えられる. 既存の文字入力方式の一部は, 母音選択の操作方向 (フリック方向など) が選択した子音によって変わってしまうため, 完全に習熟するまでは母音の選択に若干のタイムラグが生じてしまうが, QuadKey はどの子音の場合でもキーの中央タッチアップで「あ段」, 左フリックで「い段」, 上フリックで「う段」, 右フリックで「え段」, 下フリックで「お段」と共通のため子音によって母音選択のタイムラグが生じにくいと考えられる.

7. おわりに

スマートウォッチ向けのかな文字入力方式としてキーの数を 4 つに限定したソフトウェアキーボードである QuadKey を提案した. QuadKey はタッチダウンしたキーから隣接キーの中央に指をスライドすることで入力状態の文字群を切り替える仕組みを備えることにより, わずか 4 キーで 60 文字の入力が可能であり, 文字を 1 ストロークで入力できるという特徴を持つ. キーの数を 4 つにすることで各キーを大きくし, オクルージョン・ファットフィンガー問題の改善を図り, 1 ストロークで入力できることにより高い入力性能を有すると考えた. 評価実験の結果, QuadKey はオクルージョン・ファットフィンガー問題を改善することができ, 入力速度が 79.7 CPM, Cerr が 5.78% となり, スマートウォッチの文字入力方式として十分実用的であることが分かった.

今後の課題として, QuadKey は実験 30 日目の時点で十分な習熟ができていない可能性が高いため, 追加の実験を行い習熟時の入力速度とエラー率を検証する必要がある. また, より効率的な文字の割当て, 視覚フィードバックの方法の検証を行う必要がある.

参考文献

- [1] Siek, K., Rogers, Y. and Connelly, K.H.: Fat finger worries: How older and younger users physically interact with PDAs, *INTERACT 2005*, pp.267–280 (2005).
- [2] Vogel, D. and Baudisch, P.: Shift: A Technique for Operating Pen-based Interfaces using Touch, *Proc. 25th Annual ACM Conference on Human Factors in Computer Systems (CHI '07)*, pp.657–666 (2007).
- [3] Google LLC: Google 手書き入力 (オンライン), 入手先 (<https://www.google.com/intl/ja/inputtools/services/features/handwriting.html>) (参照 2018-12-06).

- [4] Apple Inc.: Apple Watch Scribble (online), available from <https://support.apple.com/ja-jp/ht206907> (accessed 2018-12-06).
- [5] Wobbrock, J., Myers, B. and Kembel, J.: EdgeWrite: A Stylus-Based Text Entry Method Designed for High Accuracy and Stability of Motion, *Proc. ACM UIST '03*, pp.61–70 (2003).
- [6] 松村 駿, 木下雄一朗, 郷健太郎: 日本語 EdgeWrite のスマートウォッチへの実装と評価, 第 20 回電子情報通信学会東京支部学生会研究発表会, p.9 (2015).
- [7] FK2 LLC: FlickKey Mini Keyboard (online), available from <http://www.flickkey.com> (accessed 2018-12-06).
- [8] ETAOI Systems: 5-TILES (online), available from <http://fivetiles.com> (accessed 2018-12-06).
- [9] 尾崎尚人, 本多信吾, 田中敏光, 秋田光平, 佐川雄二: ストロークジェスチャとタップの組み合わせによる角形スマートウォッチ向けの文字入力手法, 情報処理学会論文誌, Vol.59, No.8, pp.1511–1519 (2018).
- [10] Oney, S., Harrison, C., Ogan, A. and Wiese, J.: ZoomBoard: A Diminutive Qwerty Soft Keyboard Using Iterative Zooming for Ultra-small Devices, *Proc. SIGCHI Conference on Human Factors in Computing Systems (CHI '13)*, pp.2799–2802 (2013).
- [11] Leiva, L.A., Sahami, A., Catala, A., Henze, N. and Schmidt, A.: Text Entry on Tiny QWERTY Soft Keyboards, *Proc. 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15)*, pp.669–678 (2015).
- [12] 石井 晃, 箱田博之, 志筑文太郎, 田中二郎: Flickey: 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 (2015).
- [13] Hong, J., Heo, S., Isokoski, P. and Lee, G.: SplitBoard: A Simple Split Soft Keyboard for Wristwatch-sized Touch Screens, *Proc. 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15)*, pp.1233–1236 (2015).
- [14] Chen, X.A., Grossman, T. and Fitzmaurice, G.: Swipeboard: A Text Entry Technique for Ultra-small Interfaces That Supports Novice to Expert Transitions, *Proc. 27th Annual ACM Symposium on User Interface Software and Technology (UIST '14)*, pp.615–620 (2014).
- [15] touchone: TouchOne Keyboard (online), available from <http://www.touchone.net/> (accessed 2018-12-06).
- [16] Whirlscape Inc.: MinimumKeyboard (online), available from <http://minuum.com/minuum-on-smartwatch/> (accessed 2018-12-06).
- [17] 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, 情報処理学会, インタラクション 2016, pp.701–703 (2016).
- [18] Tojo, T., Kato, T. and Yamamoto, S.: BubbleFlick: Investigating Effective Interface for Japanese Text Entry on Smartwatches, *Proc. ACM MobileHCI '18*, pp.44:1–44:12–70 (2018).
- [19] 東條貴希, 加藤恒夫, 山本誠一: BubbleSlide: フリック操作の規則性を高めたスマートウォッチ向け日本語かな入力インタフェース, 情報処理学会, インタラクション 2018, pp.1048–1053 (2018).
- [20] 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).
- [21] 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. Mobile*

HCI 2013-Touch and Text Input, pp.161–179 (2013).

- [22] 国立国語研究所コーパス開発センター: 「現代日本語書き言葉均衡コーパス」語彙表短単位語彙表データ (オンライン), 入手先 http://pj.ninjal.ac.jp/corpus_center/bccwj/freq-list.html (参照 2018-12-06).
- [23] Soukoreff, R.W. and Mackenzie, I.S.: Metrics for text entry research: An evaluation of MSD and KSPC, and a new unified error metric, *Proc. SIGCHI Conf. Human Factors in Computing Systems (CHI '03)*, pp.113–120 (2003).



安福 和史

2019 年日本大学大学院生産工学研究科博士前期課程修了。在学中, 小型端末における文字入力方式の研究に従事。ヒューマンインタフェース学会会員。



中村 喜宏

1991 年徳島大学大学院工学研究科修士課程修了。同年日本電信電話 (株) 入社。以来, ユーザインタフェース, 知的教育システム, ネットワークオペレーションシステムの研究に従事。2007 年日本大学生産工学部数理情報工学科准教授, 2015 年同大学教授, 現在に至る。ヒューマン・コンピュータ・インタラクション等の研究に従事。電子情報通信学会, ヒューマンインタフェース学会各会員。博士 (工学)。