

クロッシングによる選択を用いた表示面積の小さい 仮想現実向け1次元キーボード

藤田 俊^{*1} 志築 文太郎^{*2}

A One-Dimensional Keyboard with Small Display Area Leveraging Crossing-Based Selection for Virtual Reality

Shun Fujita^{*1}, Buntarou Shizuki^{*2}

Abstract – A conventional QWERTY keyboard with ray-cast key selection (conventional method), which is often used for text entry in virtual reality, requires a large occupying display area. In this study, we developed CrossBoard, a one-dimensional keyboard for virtual reality. The keys are arranged densely in a single row to reduce the occupying display area, which are selected by ray-cast crossing; that is, a key is selected when the ray cursor crosses the key. In addition, we designed its word prediction function based on the results of preliminary experiments on the key selection characteristics (spatial model) of CrossBoard and its optimal size of the keyboard. A comparative user study with the conventional method shows that the performance of CrossBoard is 67.3% of the conventional method, even though CrossBoard occupies the display area 26.0% smaller than the conventional method.

Keywords : text entry, ray-cast, crossing, qwerty keyboard

1. はじめに

現在、Head-mounted Display (HMD) に基づく仮想現実における文字入力手法として、QWERTY 配列のソフトウェアキーボードを仮想空間内に表示し、ユーザがキーをレイキャストによるポインティングにて選択するという手法（従来手法）が普及している。ただし、ソフトウェアキーボードを用いた文字入力では、ソフトウェアキーボード、および入力途中の文字列（例：Web ブラウザのアドレスバー）の両者をユーザが見る必要があり^[1]、これにともなって両者を近くに配置することが望ましい。しかしながら、従来手法におけるキーボードは表示に要する面積が大きい。そのため、両者を近くに配置すると、入力途中の文字列の周囲にある表示要素（例：Web ブラウザ全体）とソフトウェアキーボードが重なる場合が多い。一方、周囲にある表示要素とソフトウェアキーボードとを重ならないように配置すると、両者が離れる。しかし、仮想現実の視野は現実世界の視野よりも狭いため^[2]、離れたものを見比べる際に視線のみならず頭部を動かす必要が生じるので不快感につながる^[3]可能性がある。

我々は仮想現実におけるキーボードの面積を抑えるために、クロッシングによる選択を用いた1次元キーボードによる文字入力手法として CrossBoard を開発

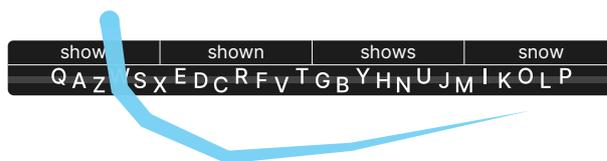


図1 CrossBoard. キーを1列に並べ、キー選択にクロッシングを採用することにより表示に要する面積を抑えている。

Fig.1 CrossBoard. It Arranges the Keys in a Single Row and Uses Crossing for Key Selection to Reduce the Occupying Display Area.

した（図1）. CrossBoardでは、キーボードの面積を抑えるために各キーを一列に密集して並べた。文字入力の際、これらのキーはレイキャストによるクロッシング、すなわちレイキャストによるカーソルがキーを横切ることにより選択される。以降では、まず CrossBoard の設計を述べる。その後、キー選択の特性およびキーボードの最適な大きさに関する予備実験（実験1）、および予備実験の結果に基づく単語予測機能を述べた後に、従来手法との比較実験（実験2）を述べる。

2. 関連研究

CrossBoard と類似する操作を行う一筆書き文字入力手法、および曖昧キーボード、仮想現実における文字入力手法と本研究の位置づけを示す。

2.1 一筆書き文字入力手法

一筆書き文字入力手法とは、ユーザが1本の軌跡を描くことによって文字あるいは単語が入力できる手法である。

*1: 筑波大学 コンピュータサイエンス専攻

*2: 筑波大学 システム情報系

*1: Department of Computer Science, University of Tsukuba

*2: Faculty of Engineering, Information and Systems, University of Tsukuba

1本の軌跡によって1文字の入力を可能とする手法が開発されている。Graffiti^[4]は、アルファベットの形を模した一筆書きの軌跡を描くことによって文字入力が行える手法である。EdgeWrite^[5]は、運動障害者向けに設計された文字入力手法であり、Graffitiと同様にアルファベットを元とする軌跡にて文字入力を実現している。ただしEdgeWriteでは、軌跡の形状ではなく軌跡とペン入力領域の枠との衝突位置に基づく。枠との衝突に基づき文字入力を行う。これらの手法は1つの軌跡によって1文字の入力が可能である。

一方、1本の軌跡によって1単語の入力を可能とする手法が知られている。Cirrin^[6]では、ユーザは環状に配置されたキーを次々となぞることにより、単語を入力できる。なお、この環状の配置は、タイル状のキー配置にて起こる単語を構成する文字以外のキーをなぞってしまう問題を防ぐための工夫である。SHARK²^[7]では、ユーザはタイル状のキー配置において入力したいキーを次々となぞることにより単語単位の入力ができる。これはなぞられたキーをそのまま文字として入力せずに、軌跡の形状からユーザが入力した単語を推定することにより実現されている。

CrossBoardは、単語単位の入力が可能な一筆書き文字入力手法のうち、軌跡の形状ではなく軌跡とキーボードとの交差に基づく文字入力手法である。ユーザが入力した単語はキーボードと交差した位置から推定される。

2.2 曖昧キーボード

曖昧キーボードとは、ユーザの多義的な入力を解消することに基づく文字入力手法である。1Line Keyboard^[8]は、タブレットにおけるソフトウェアキーボードの表示面積を抑えるために、QWERTY配列に基づくキーの統合を行った手法である。統合されたキーには複数の文字が含まれている（例えば、“QAZ”が1つのキー）。

一方、キーを削減せずにユーザの不正確なキー選択を許容する曖昧キーボードも存在する。1次元キーボードにおいて、傾き^[9]、タッチ^[10]、ならびに押下圧^[11]による不正確なキー選択を許容する手法が存在する。また、通常のスマートフォンのソフトウェアQWERTYキーボードにおいて、キーそのものを表示しない状況における不正確なキー選択を用いた手法^[12]が存在する。この種の曖昧キーボードでは、どのキーが選択されたのかを表現したモデル、および言語モデルと呼ばれる単語の出現頻度を表現したモデルを用いて単語予測を行う。

曖昧キーボードは限られた表示面積^{[8],[12]}あるいは入力手段^{[9],[11]}における文字入力の実現に効果的なため、スマートウォッチにおける文字入力手法^{[13],[14]}に

も応用されている。

CrossBoardはクロッシングによる不正確なキー選択を許容する手法^{[9]~[11]}であるとともに、表示面積を抑えたキーボード^{[8],[12]}による文字入力を実現する。

2.3 仮想現実における文字入力

仮想現実における文字入力には、物理キーボード^[15]、手の動き^{[16]~[19]}、ならびにコントローラ^{[20]~[22]}など様々な操作が用いられてきた。

Walkerら^[15]は、仮想現実においても物理キーボードを用いた文字入力において、入力されたキーをユーザにフィードバックし、加えて入力を自動補正することにより、現実での文字入力と同等の性能を発揮できることを示した。

手の動きによる文字入力手法には、物理キーボードにおける文字入力を模した手法^{[18],[19]}、スワイプキーボードを模した手法^[16]、1Line Keyboard^[8]を模した手法^[17]がある。採用されている手の動きとして、ボタンを押し込む動き^{[18],[19]}、指にてつまむ動き^{[16],[17]}、および指にてつまみそのまま空中を描く動き^{[16],[17]}がある。

コントローラを用いた文字入力手法については比較調査^{[20],[21]}がおこなわれており、結果として、レイキャストを用いた手法（本稿における従来手法）^[20]およびコントローラから伸びた仮想的なスティックの先端にてキーを叩くことにより文字入力する手法^[21]が有望であることが示されている。なお、この2手法はコントローラそのものを動かすことによる手法であるが、コントローラ上のジョイスティックのみを用いた手法^[22]も存在する。PizzaText^[22]は、2つのジョイスティックの入力の組み合わせを用いた、仮想現実における表示面積の小さい文字入力手法である。

CrossBoardはコントローラの動きを用いた文字入力手法であり、文字入力に用いるキーボードの表示面積を小さくすることを目的として開発されている。

3. CrossBoard

本節では、CrossBoardの設計および操作方法を述べる。

3.1 設計

CrossBoardは、キーボードの表示面積を通常のQWERTYキーボードよりも小さくすることを目的とした手法である。この目的の実現のために、CrossBoardのキーボード配列、キー選択手法、および入力デバイスを以下の通りに設計した。

キーボード配列 学習コストを低減するためにQWERTY配列を元とした配列（図1）を採用した。この配列はQWERTY配列を構成するキーを左から1列ずつ取り出して並べたもの

(QAZWSX...)であり、列ごとにガタつかせている。QWERTY配列を採用した理由は、Rotext^[9]にて、視覚フィードバックのある条件下においてQWERTY配列のパフォーマンスがユーザにとって未知の配列に比べて優れていたためである。また、列ごとにガタつかせた配列とした理由は、キーを探しやすくするためであり、またガタつかせた配列の性能の方がそうでない配列よりも良い傾向が示されている^[9]ことによる。

キー選択手法 レイキャストによる高速な選択を可能とするためにクロッシング^{[23],[24]}を採用した。レイキャストによる入力にはコントローラを用いた操作においてよく用いられているため、これを採用すれば文字入力と他のGUIの操作との一貫性を持たせられる。また、クロッシングには、画面専有面積を抑えられる画面設計を可能とするというメリットが示されている^[25]。

入力デバイス キー選択には利き手に把持されたコントローラを用いる。これは軌跡による文字入力において両手入力を採用しても入力性能は向上せず、また知的・知覚的負荷が増加する^[26]ことが知られていたためである。

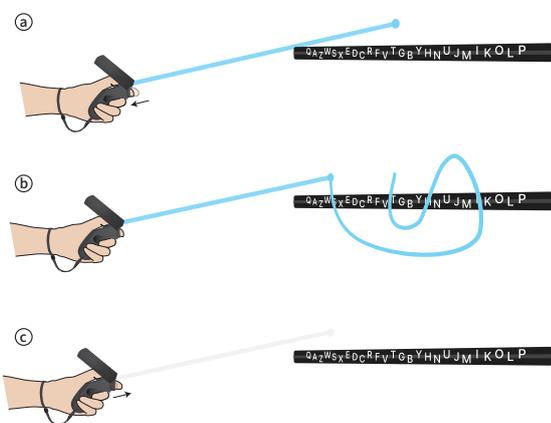


図2 単語入力。ここでは“this”を入力している。
a: ユーザはまずコントローラのトリガボタンを引く。b: ユーザはクロッシングにて‘t’, ‘h’, ‘i’, ‘s’の順にキーを選択する。c: ユーザはトリガボタンを離す。トリガボタンを離すことにより単語の予測変換が行われる。

Fig. 2 Word Input. Here, the User Is Typing “this”. a: The User First Pulls the Trigger Button on the Controller. b: The User Selects the Keys ‘t’, ‘h’, ‘i’, and ‘s’ With Crossing. c: The User Releases the Trigger Button. The Word Prediction Is Performed When the Trigger Button Is Released.

3.2 操作方法

単語入力の手順を図2に示す。ユーザは単語を入力する際、まずコントローラのトリガボタンを引く(図2a)。トリガボタンを押している間に、単語を構成する文字に対応するキーにカーソルをクロスさせることにより、文字入力を行う(図2b)。カーソルがキーをクロスした時、ユーザには打鍵した音を模した音声、コントローラの振動、およびキーボード水平方向の線がフラッシュするという視覚効果が同時にフィードバックされる。なお、キーの選択の際、ユーザは正確にキーにカーソルをクロスさせる必要はなく、おおよその位置で良い。単語を構成する文字を入力し終えたのちユーザがトリガボタンを離したとき、単語の予測変換が行われる(図2c)。

単語の削除は、入力済みの単語を塗りつぶす操作にて行われる(図3)。ユーザは削除したい単語の上にカーソルを合わせ、トリガボタンを引き、そのまま削除したい範囲をカーソルにて3往復以上塗りつぶす(図3b)。このとき塗りつぶされた範囲に応じて単語削除が行われる範囲は変化する。ユーザがトリガボタンを離すと、塗りつぶされた範囲に含まれる単語が削除される(図3a)。なお、単語の削除を行ったとき、単語間の空白は英文として自然な形に修正される。

4. 実験1: 最適なキーボードの大きさの調査

CrossBoardにおける各キーについて、ばらつきなどの選択位置の特性(以降、空間モデル)はキーボードの大きさによって変化する可能性がある。また、CrossBoardのキーボードの大きさがユーザの好みに影響する可能性も考えられる。そこで、キーボードの大きさによる空間モデルの変化およびキーボードの大きさによる

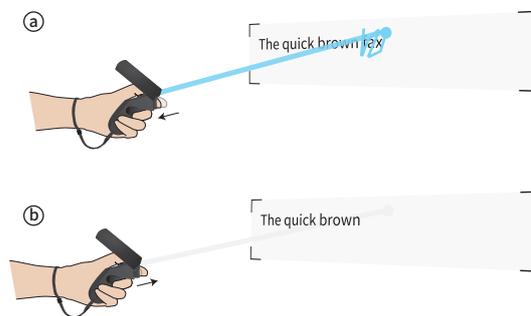


図3 単語の削除。a: ユーザはトリガボタンを引いたまま削除したい範囲を3往復以上塗りつぶす。b: ユーザがトリガボタンを離すと、塗りつぶされた単語が削除される。

Fig. 3 Deleteing Words. a: The User Pulls the Trigger Button and Scribble Through the Text to Be Deleted at Least Three Times. b: When the User Releases the Trigger Button, the Words Scribbled Will Be Deleted.

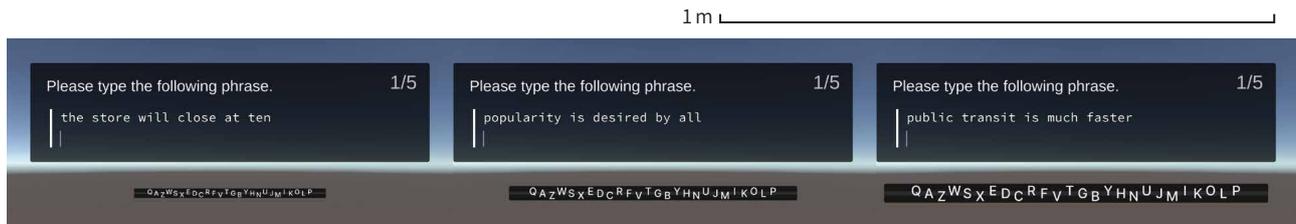


図4 実験1にて比較したキーボード. 左: Small. 中央: Medium. 右: Large.
 Fig. 4 Keyboards Compared in Experiment 1. Left: Small, Center: Medium,
 Right: Large.

関する好みを調査した。

4.1 実験環境および参加者

仮想現実用 HMD として Oculus Quest (視野角: 94 度 × 94 度^[27], 単眼解像度: 2880 px × 1600 px) を用いた。参加者はタスクを行っている間, 起立し両手にコントローラを把持していた。仮想空間には, キーボード, 提示文, 入力フォームが表示される GUI 平面があり, GUI 平面を参加者から 1 m 離れた位置に参加者の目線の高さにて表示された。

筆者が所属する大学の大学生および大学院生の 15 名 (年齢 21–24 歳: $M = 22.80, SD = 1.08$, 男性 12 名, 右利き 12 名, 眼鏡あり 6 名) を参加者とした。全ての参加者は普段から QWERTY キーボードを使用している。

4.2 タスクおよび手順

タスクは英文文章を転写するタスクであり, 参加者は提示された英文を CrossBoard のモックアップを用いて書き写した。参加者がキーを選択すると, 入力フォームには文字のかわりに ‘-’ が入力されるため, 参加者は自身が入力したキーを知ることはできない。ただし, 入力した単語の削除は許容されており, ミスタイプしたと感じた単語の再入力が可能である。このとき, 参加者には特定の文字にこだわることなく素早く正確に入力するように指示した。

実験には横幅 34.62 cm (Small), 51.92 cm (Medium), 69.23 cm (Large) の 3 種類のキーボードを用いた (図 4)。それぞれのキーボードの各キーの文字の相対的な大きさおよび位置関係は同一であり, キーボードの両端には余白領域がそれぞれ 2 キー分設けられている。

1 つのキーボードにて英文 50 文を転写することを 1 セッションとした。1 セッションにおいて, 参加者には MacKenzie ら^[28] の英文セットからランダムに抽出した 50 文を入力してもらった。1 セッションは 1 回の練習ブロックと 4 回の本番ブロックから構成されており, 各ブロックにおいて 10 文ずつ参加者に提示した。セッションの実施順序は参加者毎にラテン方格にてカウンタバランスをとった。練習セッションも含

め, 実験を通じて 15 名 × 3 条件 × 50 文 = 2250 文の入力を収集した。

全セッション終了後に, 参加者が使いやすいと感じたキーボードの大きさをフリーハンドにて描いてもらい収集した。

実験の所要時間は, 事務手続きも含め約 120 分であった。実験終了後, 参加者は謝金として 1720 円を受けとった。

4.3 結果および考察

本節では, 実験結果およびその考察を述べる。

4.3.1 空間モデル

ここでは, キー選択がどのように行われていたかを議論する。以降, キーの選択位置に関して論じる際の長さの単位として Key を用いる。Key は各大きさ条件における 1 つのキーが占める幅である。1 Key は, Small において 1.154 cm, Medium において 1.731 cm, Large において 2.308 cm となる。

なお, 単語の綴り間違いによるミスタイプ (例えば, ‘a’ を ‘o’ と ‘u’ に, ‘e’ を ‘u’ に, ‘r’ を ‘l’ に間違えるなど) が観察された。これを外れ値とするため, キーの中心から 4 Key 以上離れたキー選択を外れ値として解析対象から除いた。

ここで, 各キーの選択位置が各キーの中心からどれだけ離れていたのかを「ずれ」と呼ぶことにする。キーボードの大きさ条件ごとの各キーの選択位置のずれの平均を図 5 に示す。独立変数をキーおよびキーボードの大きさ, 従属変数をキーの選択位置として反復測定二元配置分散分析を行ったところ, キーによる主効果が存在し, キーボードの大きさによる主効果はなく, また交互作用が存在した (キー: $F_{25,342} = 3.429, p < .001$, キーボードの大きさ: $F_{2,40} = 1.420, p = .254$, 交互作用: $F_{50,51063} = 2.848, p < .001$)。図 5 を確認すると, 全体的にずれは右下がりの傾向にあることから, キーの選択はキーボードの中心方向にずれる傾向にあったことがわかる。また, キーボード中央部 (R キーから N キー) について, 3 キー周期のキー選択位置のずれの変化と, 端のキー (Q キーおよび P キー) における全体の右下がりの傾向から逸脱する傾向が確認さ

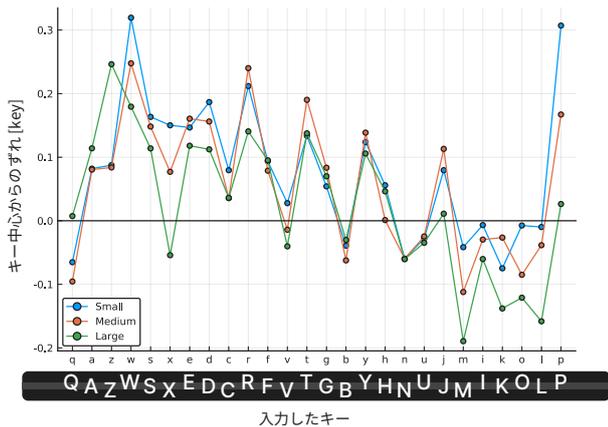


図5 キーボードの大きさごとの各キーの選択位置のずれ。縦軸正方向はキーボード右方向へのずれを意味する。

Fig. 5 The Offset of the Selected Position of Each Key in the Size of the Keyboard. The Positive Offset Indicates Hitting to the Right of the Center of Key.

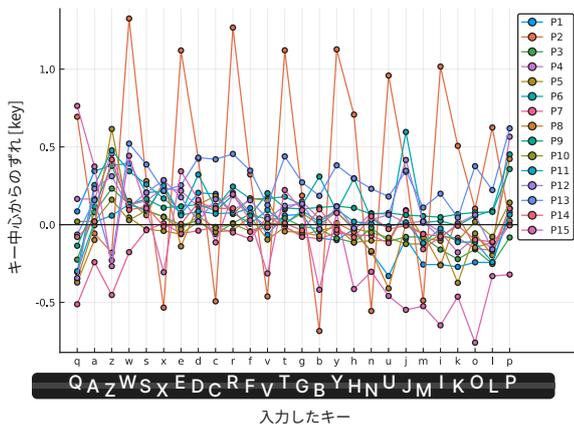


図6 参加者ごとの各キーの選択位置のずれ。縦軸正方向はキーボード右方向へのずれを意味する。

Fig. 6 The Offset of the Selected Position of Each Key in Each Participant. The Positive Offset Indicates Hitting to the Right of the Center of Key.

れた。これは、キーボード中央部についてはキーボードのキーレイアウトの変化と一致しており、キーの縦列については中列のキーがある方向にキー選択のずれが発生することがわかる。端のキー選択のずれについてはキーボードの左右両端の空白領域方向へずれたものである。このことから、キーボードの視覚設計が、ユーザのキーの選択位置に影響を与えたことが示唆される。

参加者ごとの各キーの選択位置のずれを図6に示す。図6より視覚設計の影響が、参加者によって異なることがわかる。なお、P2は他の参加者と比べて視覚設計の影響を大きく受けており、1Keyを超えるずれがキーボードのレイアウトと一致するように発生してい

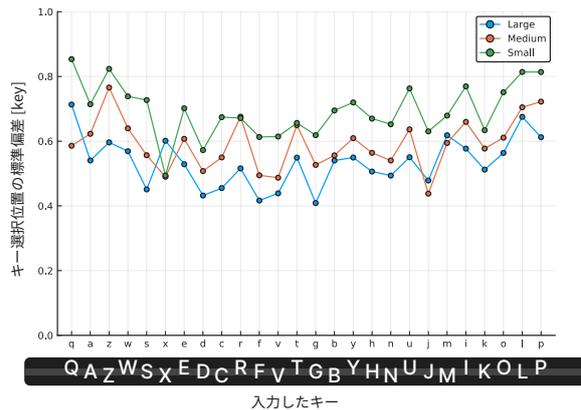


図7 大きさ条件ごとの各キーの選択位置のばらつき。

Fig. 7 The Variance of the Selected Position of Each Key in the Size of the Keyboard.

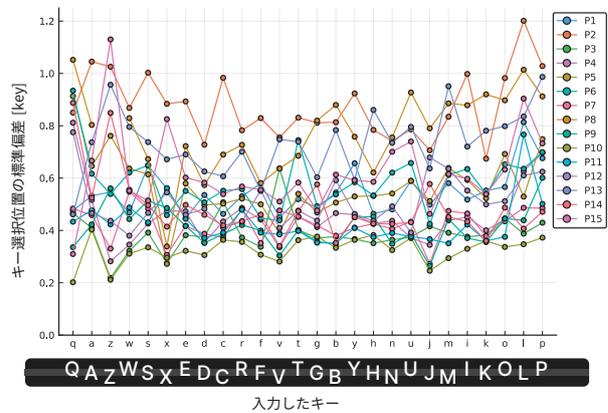


図8 参加者ごとの各キーの選択位置のばらつき。

Fig. 8 The Variance of the Selected Position of Each Key in Each Participant.

る。P2は「今のキーボードの形だと、QWERTYのタテ3文字ずつで区切られている印象で実験中もそれを元に入力を行った」と報告している。P2については、QWERTY配列の縦列1列をひとまとまりと見なしてタイプしていたと考えられ、これにより各列の中心のキーに選択位置が偏ったと考えられる。

次に、各キーの選択位置のずれを基準とした、キー選択の標準偏差を「ばらつき」と呼ぶことにし、これを分析する。キーボードの大きさについてばらつきにどのような差があるのかを調べるために、キーボードの大きさごとの各キーの選択位置のばらつきを図7に示す。独立変数をキーおよびキーボードの大きさ、従属変数を各キーごとのキー選択の標準偏差として反復測定二元配置分散分析を行ったところ、キーおよびキーボードの大きさについて主効果が存在した (Greenhouse-Geisserの補正を用いたもの、キー: $F_{25,299.27} = 5.249, p < .001$, キーボードの大きさ: $F_{2,28.31} = 26.40, p < .001$)。このことから、キーボー

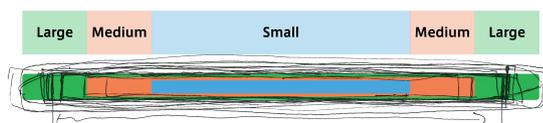


図9 参加者が使いやすいと感じたキーボードの大きさの手書きの重畳表示.

Fig. 9 Merged Handwriting of the Size of the Keyboard That the Participants Felt Easy to Use.

ドが大きくなるにつれて相対的にキー選択のばらつきが小さくなっていること、キーによって選択精度が異なることがわかる。

参加者ごとの各キーの選択位置のばらつきを図8に示す。図8から、参加者ごとのばらつきの差の影響がキーボードの大きさよりも大きいことがわかる。参加者によってキー選択のばらつきに差がある原因として、「特定の文字にこだわることなく素早く正確に入力するように」という指示に対して各参加者が受ける影響に差があり、この差によって選択の素早さと正確さの優先度に差があった可能性が考えられる。

4.3.2 使いやすいと感じた大きさ

各条件のキーボードについて参加者に使いやすいキーボードの大きさを手書きしてもらった結果を図9に示す。参加者が使いやすいと感じたキーボードの大きさの平均は61.07 cm (SD = 6.892 cm)であり、これはMediumとLargeの間の大きさにあたる。

このことから、キーボードの大きさとして61.07 cmを以降の実験にて採用する。

5. 単語予測

CrossBoardにおける各キーの選択はおおよその位置にて行われるため、ユーザが入力しようとした単語の予測が必要となる。CrossBoardでは^{[11],[13],[14]}にて採用されている手法を参考に単語予測を行う。候補単語 $w = \{l_1, l_2, \dots, l_m\}$ (l_i は文字、候補単語の長さ m は $n \leq m$ を満たす)とした時、クロッシングした座標列 $X = \{x_1, x_2, \dots, x_n\}$ に対して、候補単語 w_{pred} を次式にて求める。

$$w_{pred} = \arg \max_{w \in L} \left(p(w) \cdot \prod_{i=1}^n p(x_i | l_i) \cdot \alpha^{m-n} \right) \quad (1)$$

ここで、 $p(w)$ は単語の出現頻度を表現する言語モデル、 $p(x_i | l_i)$ はキー l_i を選択した時に x_i の位置が選択されている確率(空間モデル)、 α^{m-n} (ただし、 $0 \leq \alpha < 1$)は、ユーザのキー入力が少ない時に出現確率が高く、文字数の多い単語の尤度を抑えるペナルティである。

これらのパラメータについて、 $p(w)$ をOANC (Open American National Corpus)^[29]から出現頻度

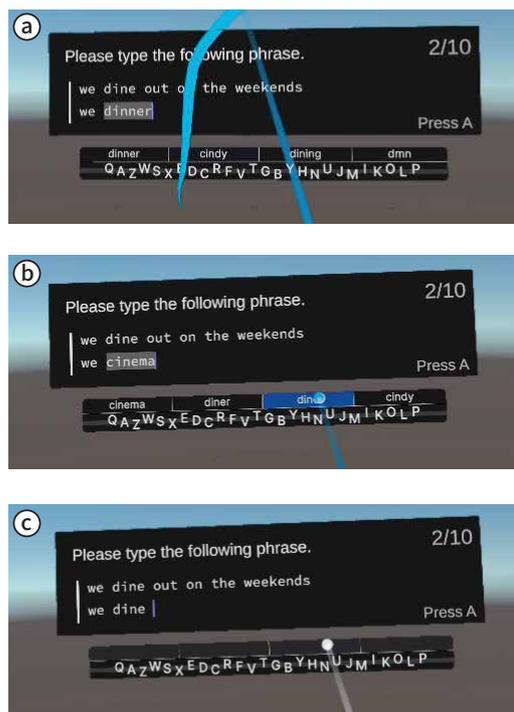


図10 CrossBoardでの入力例。ここでは“dine”を入力している。a: ‘d’, ‘i’, ‘n’, ‘e’ と入力し終わった状態。b: 候補単語から“dine”を選ぶ様子。c: “dine”が入力された状態。

Fig. 10 Example of Typing on CrossBoard. In This Example, the User Types “dine”. a: Finished Typing ‘d’, ‘i’, ‘n’, and ‘e’. b: Selecting “dine” From the Candidate Words. c: “dine” Has Been Entered.

の高い上位70000単語から、 $p(x_i | l_i)$ を実験1の結果から求め、また α を筆者1名が実験的に値の調整を行い $\alpha = 0.6$ に設定した。

単語予測機能を実装したCrossBoardにおける文字入力の様子を図10に示す。このCrossBoardには実験1にて使用したキーボードの上部に候補単語リストを表示する領域を追加した。候補単語は左から第1, 第2, 第3, および第4候補となっており、左に表示されている単語ほど尤度が高いものである。第1候補の候補単語は自動的に入力欄に書き込まれ、ユーザがそのまま次の単語入力を行った場合、第1候補の単語が自動的に入力される。第1候補の候補単語が意図しない単語の場合、ユーザは第2から第4候補から意図した単語を探しポインティングにて選択することにより意図した単語の入力ができる(図10b, c)。

6. 実験2: 従来手法との比較

CrossBoardと従来手法との比較実験を行った。

6.1 実験環境および参加者

実験環境は実験1と同一である。

参加者は、研究室内の大学生および大学院生の8名

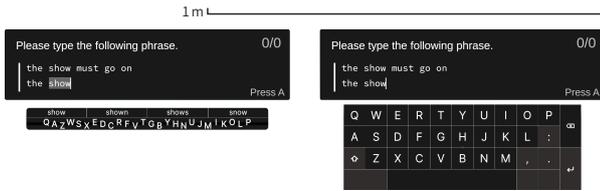


図 11 実験 2 にて比較したキーボード. 左: Cross-Board. 右: 従来手法 (2D Keyboard).

Fig. 11 Keyboards Compared in Experiment 2. Left: CrossBoard. Right: Conventional Method (2D Keyboard).

(年齢 22-25 歳: $M = 23.25, SD = 1.16$, 男性 7 名, 右利き 7 名, 眼鏡あり 4 名) を参加者とした. 参加者全員が普段から QWERTY キーボードを使用している.

6.2 タスクおよび手順

タスクは英文文章を転写するタスクであり, 参加者は提示された英文を CrossBoard あるいは従来手法を用いて書き写した. 参加者には, 可能な限り素早く正確に文字入力するように指示した.

実験にて使用したキーボードを図 11 に示す. 本実験では CrossBoard と従来手法 (2D Keyboard) の 2 種類のキーボードをそれぞれタスクに用いた. キーボードの大きさは CrossBoard について $61.07 \times 5.57 = 340.2 \text{ cm}^2$, 2D Keyboard について $60.00 \times 21.82 = 1309 \text{ cm}^2$ としており, CrossBoard は 2D Keyboard の 26.0% の面積となっている. 2D Keyboard は Oculus Quest v23 における英語キーボードのレイアウトおよび挙動を再現したものであるが, ただし元のものと異なりキーボードの湾曲を除き平面的にし, コントローラの振動によるフィードバックを追加した. これは, 2D Keyboard の細かな振る舞いを CrossBoard に近づけることを意図したものである.

実験 1 と同じく, 1 つのキーボードにて英文 50 文を転写することを 1 セッションとした. 1 セッションにおいて, 参加者には MacKenzie ら^[28] の英文セットからランダムに抽出した 50 文を入力してもらった. 1 セッションは 1 回の練習ブロックと 4 回の本番ブロックから構成されており, 各ブロックにおいて 10 文ずつ参加者に提示した. セッションの実施順序は参加者毎にカウンタバランスをとった. 練習セッションも含め, 実験を通じて 8 名 \times 2 条件 \times 50 文 = 800 文の入力を収集した.

各セッション終了後に, SUS (System Usability Scale) によってユーザビリティ, および 10 段階のリッカート尺度による 1 対比較を省略した NASA-TLX にて作業負荷を測定したのち, 手法に関するコメントを求めるアンケートを実施した. 実験の所要時間は, 約 90 分であった.

6.3 評価指標

評価指標として, 単語入力速度 (WPM: Words Per Minute) および未修正エラー率 (UER: Uncorrected Error Rate)^[30] を用いた.

6.4 結果および考察

本節では, 実験結果およびその考察を述べる.

6.4.1 入力速度

練習ブロックを除くブロックにおける各手法における文ごとの平均文字入力速度は, CrossBoard について 12.86 WPM ($SD = 4.16$), 2D Keyboard について 19.11 WPM ($SD = 4.23$) であった. ブロック間における手法ごとの文字入力速度を図 12 に示す. 独立変数を手法およびブロック, 従属変数を入力速度として反復測定二元配置分散分析を行ったところ, 手法およびブロック間について主効果が存在した (手法: $F_{1,7} = 81.01, p < .001$, ブロック: $F_{4,28} = 32.58, p < .001$).

参加者ごとに見た場合, 両手法について P5 が最も文字入力速度が高く CrossBoard では第 4 ブロックにおいて平均 16.58 WPM, 2D Keyboard では第 3 ブロックにおいて平均 25.36 WPM を記録している.

6.4.2 エラー率

練習ブロックを除くブロックにおける各手法における文ごとの未修正エラー率は, CrossBoard について 1.37% ($SD = 7.48$), 2D Keyboard について 1.30% ($SD = 6.70$) であった. ブロック間における手法ごとの未修正エラー率を図 13 に示す. 独立変数を手法およびブロック, 従属変数を入力速度として反復測定二元配置分散分析を行ったところ, ブロック間について主効果が存在した (ブロック: $F_{4,31.54} = 3.53, p < .05$) であった. ブロック間における未修正エラー率の低減は確認されたものの, 手法間の未修正エラー率について差があるとはいえなかった.

6.4.3 SUS

SUS の得点はそれぞれ, CrossBoard は平均 60.94 ($SD = 22.08$), 2D Keyboard は平均 83.13 ($SD = 5.79$) であった. 各手法の得点は Bangor らの Letter Grade Scale^[31] (A から F までの評価, A が最もよい) に読み替えると, CrossBoard は D 評価, 2D Keyboard は B 評価に該当する. 手法間のスコアについてウェルチの t 検定を行ったところ, 差が示された ($t_{7.96} = 2.75, p < .05$).

6.4.4 NASA-TLX

NASA-TLX の各下位尺度 (MD, PD, TD, OP, EF, および FR) および平均作業負荷 (WL) の箱ひげ図を図 14 に示す. それぞれの尺度について対応のある t 検定を行ったところ, MD について差が示された ($t_7 = 4.07, p < .01$). また, PD を除くすべての尺度について CrossBoard の負荷が高い傾向が確認さ

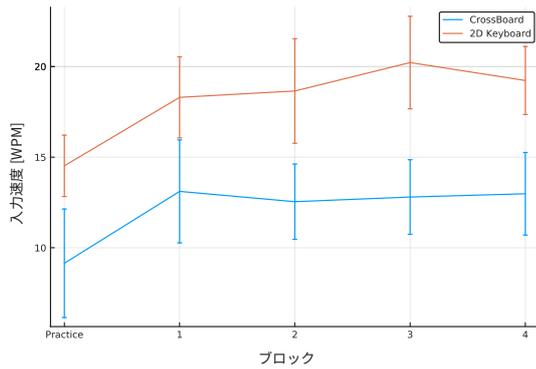


図 12 ブロック間における手法ごとの文字入力速度.

Fig. 12 Text Entry Rate for Both Keyboard for Each Block.

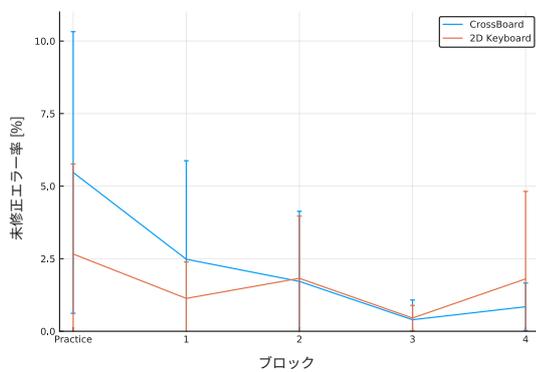


図 13 ブロック間における手法ごとの未修正エラー率.

Fig. 13 Uncorrected Error Rate for Both Keyboard for Each Block.

れた.

6.4.5 コメント

参加者から単語予測機能に関するコメントが多く挙げられた。3名の参加者から、候補単語の文字数が入力した文字数と一致しないため、入力中にどこまでタイプしたかを忘れるというコメントを得た。1名の参加者から、キー入力が少しでも外れると適切に単語予測が効かないため、よりずれたキーを選択した場合でもうまく単語予測して欲しいというコメントがあった。また、類似するコメントとしてトリガボタンを離すと候補単語の選択のみしかできなくなるため、意図した単語が候補単語にない場合に単語を削除し入力し直す必要があったことが1名の参加者から報告されている。

7. 議論および今後の課題

本節では全体を通じた議論を行うとともに今後の課題を述べる。

7.1 応用例

実験2により、CrossBoardは、従来手法に比べて表示に要する面積が26.0%であるにも関わらず、67.3%

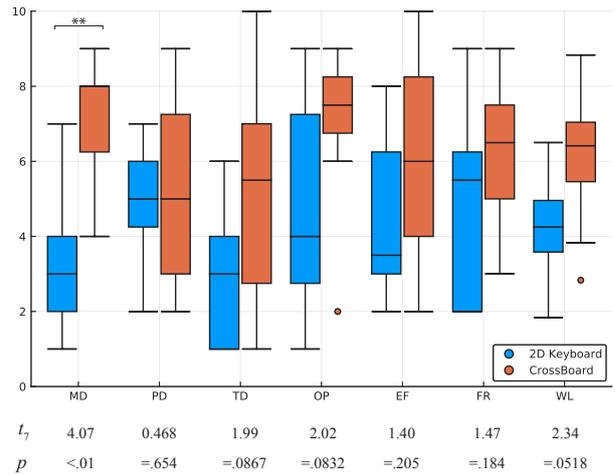


図 14 手法ごとの NASA-TLX の各下位尺度および作業負荷. MD: 知的・知覚的要求, PD: 身体的要求, TD: タイムプレッシャー, OP: 作業成績, EF: 努力, FR: フラストレーション, WL: 作業負荷 (下位尺度の平均).

Fig. 14 NASA-TLX Subscales and Workloads for Both Keyboard. MD: Mental Demand, PD: Physical Demand, TD: Temporal Demand, OP: Performance, EF: Effort, FR: Frustration, WL: Workload (Mean of Subscales).

の入力速度であることがわかった。このような入力速度の低下を加味してもキーボードの表示面積を抑えたい状況においては、CrossBoardが有用であると考えている。例えば、動画閲覧中に受信したテキストチャットに対する簡単な返信のような、主となるコンテンツを遮りたくなく多少の入力速度の低下が許容される状況において、CrossBoardは有用であると考えられる。

また、仮想現実以外にもCrossBoardを応用できる状況が考えられる。例えば、いくつかの拡張現実用HMDにおける表示視野角は、汎用的な仮想現実用HMDよりも狭い^[2]ため、CrossBoardが有用な状況が存在すると考えられる。他にも、スマートテレビの検索画面では、入力単語に応じて候補となるコンテンツを表示する場面があるが、通常のQWERTYキーボードを採用した場合、表示面積の多くをキーボードが占める。このような状況においてCrossBoardを採用することによって、候補となるコンテンツをより多く表示できると考えられる(図15)。

7.2 単語予測の改善

実験2において単語予測に関するコメントが多く寄せられたため、単語予測の精度を向上させることが今後の課題として挙げられる。単語予測の精度を改善するために以下のような方針を検討している。

先読み機能の調整 ユーザが入力した文字数と一致しない候補単語が表示されることにより、ユーザ

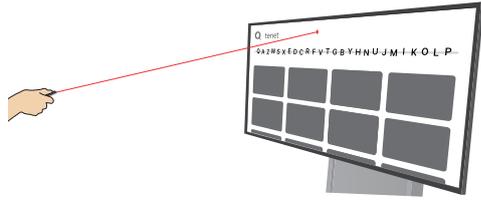


図 15 CrossBoard のスマートテレビへの応用。CrossBoard を利用することで他のコンテンツの表示領域を確保している。

Fig. 15 Application of CrossBoard to Smart TVs. By Using CrossBoard, the Display Area of Other Contents Is Widened.

がどこまで文字を入力したのかを忘れることを防ぐために、候補単語については基本的に入力された文字数と一致するもののみを表示するべきである。

エリアカーソルの導入 実験 1 において、空間モデルにおけるキー選択のばらつきが参加者によって異なることがわかった。これを改善するために、キー選択をどの程度正確に行う必要があるのかをガイドとして表示することを考えている。例えば、エリアカーソル^[32]のような「キーにカーソルが触れると」キー選択できる手法を導入することにより、どの程度正確にキー選択をすべきかをカーソルの大きさによってユーザに伝えられると考えている。

言語モデルの改善 今回採用した言語モデルは単語の出現頻度に基づくものであるため、それまでにユーザが入力した単語を考慮していない。入力単語の履歴を考慮した言語モデルを導入することにより、よりの確な候補単語が表示されることが期待できる。

今後は、上記に挙げた改善を行ったのち、入力速度、ユーザビリティ、および作業負荷に関する再調査を行う必要がある。

8. おわりに

我々はクロッシングによる選択を用いた 1 次元曖昧キーボードによる文字入力手法として CrossBoard を開発した。CrossBoard は、仮想現実向けの表示面積の小さい次元キーボードの一種であり、ユーザは各キーをクロッシングにて選択することで文字入力を行う。また各キーの選択は、単語予測機能により補完されるため、ユーザは不正確なキー選択を行っても意図した単語の入力が行える。

システムの実装にあたり、CrossBoard におけるキーボードの最適な大きさおよび単語予測に必要な空間モデルを調査を行った。実験では、空間モデルの特徴に

ついてキーボードの視覚設計が空間モデルに影響を与えている可能性を示した。また、今回収集した空間モデルについて、参加者による特性の差について議論した。またキーボードの最適な大きさとして、参加者が使いやすいと感じたキーボードの大きさを収集し、その平均は 61.07 cm であった。

この実験の結果を元に CrossBoard を実装し、通常の QWERTY キーボードと比較実験を行った。実験では、CrossBoard について 12.86 WPM、最も早い参加者は最終ブロックにて平均 16.58 WPM まで到達したものの、これらの入力速度は通常の QWERTY 配列の仮想現実向けのキーボードに比べて有意に遅い入力速度だった。CrossBoard は、従来手法に比べて表示に要する面積が 26.0% であるにも関わらず、67.3% の入力速度であることがわかった。今後は単語予測機能について改善を進めることにより、入力速度、ユーザビリティ、および作業負荷の改善を進めていく。

参考文献

- [1] Jiang, X., Li, Y., Jokinen, J. P., Hirvola, V. B., Oulasvirta, A. and Ren, X.: How We Type: Eye and Finger Movement Strategies in Mobile Typing; In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, CHI '20, ACM, pp. 582:1–582:14 (2020).
- [2] Kress, B. C.: Digital Optical Elements and Technologies (EDO19): Applications to AR/VR/MR; In *Digital Optical Technologies 2019* (Kress, B. C. and Schelkens, P., eds.), Vol. 11062, SPIE, pp. 343–355 (2019).
- [3] Yu, C., Gu, Y., Yang, Z., Yi, X., Luo, H. and Shi, Y.: Tap, Dwell or Gesture?: Exploring Head-Based Text Entry Techniques for HMDs; In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, CHI '17, ACM, pp. 4479–4488 (2017).
- [4] MacKenzie, I. S. and Zhang, S. X.: The Immediate Usability of Graffiti; In *Proceedings of the Conference on Graphics Interface '97*, GI '97, Canadian Information Processing Society, pp. 129–137 (1997).
- [5] Wobbrock, J. O., Myers, B. A. and Kembel, J. A.: EdgeWrite: A Stylus-Based Text Entry Method Designed for High Accuracy and Stability of Motion; In *Proceedings of the 16th Annual ACM Symposium on User Interface Software and Technology*, UIST '03, ACM, pp. 61–70 (2003).
- [6] Mankoff, J. and Abowd, G. D.: Cirrin: A Word-Level Unistroke Keyboard for Pen Input; In *Proceedings of the 11th Annual ACM Symposium on User Interface Software and Technology*, UIST '98, ACM, pp. 213–214 (1998).
- [7] Kristensson, P.-O. and Zhai, S.: SHARK²: A Large Vocabulary Shorthand Writing System for Pen-Based Computers; In *Proceedings of the 17th Annual ACM Symposium on User Interface Software and Technology*, UIST '04, ACM, pp. 43–52 (2004).

- [8] Li, F. C. Y., Guy, R. T., Yatani, K. and Truong, K. N.: The 1Line Keyboard: A QWERTY Layout in a Single Line; In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology*, UIST '11, ACM, pp. 461–470 (2011).
- [9] Walmsley, W. S., Snelgrove, W. X. and Truong, K. N.: Disambiguation of Imprecise Input with One-dimensional Rotational Text Entry; In *ACM Transactions on Computer-Human Interaction*, (2014).
- [10] Whirlscape: Minuum Keyboard | By Whirlscape; (2021). <http://minuum.com/>.
- [11] Zhong, M., Yu, C., Wang, Q., Xu, X. and Shi, Y.: ForceBoard: Subtle Text Entry Leveraging Pressure; In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, CHI '18, ACM, pp. 528:1–528:10 (2018).
- [12] Zhu, S., Luo, T., Bi, X. and Zhai, S.: Typing on an Invisible Keyboard; In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, CHI '18, ACM, pp. 439:1–439:13 (2018).
- [13] Yi, X., Yu, C., Xu, W., Bi, X. and Shi, Y.: COM-PASS: Rotational Keyboard on Non-Touch Smartwatches; In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, CHI '17, ACM, pp. 705–715 (2017).
- [14] Gong, J., Xu, Z., Guo, Q., Seyed, T., Chen, X. A., Bi, X. and Yang, X.-D.: WrisText: One-handed Text Entry on Smartwatch using Wrist Gestures; In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, CHI '18, ACM, pp. 181:1–181:14 (2018).
- [15] Walker, J., Li, B., Vertanen, K. and Kuhl, S.: Efficient Typing on a Visually Occluded Physical Keyboard; In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, CHI '17, ACM, pp. 5457–5461 (2017).
- [16] Markussen, A., Jakobsen, M. R. and Hornbæk, K.: Vulture: A Mid-Air Word-Gesture Keyboard; In *Proceedings of the 32nd Annual ACM Conference on Human Factors in Computing Systems*, CHI '14, ACM, pp. 1073–1082 (2014).
- [17] Fashimpaur, J., Kin, K. and Longest, M.: PinchType: Text Entry for Virtual and Augmented Reality using Comfortable Thumb to Fingertip Pinches; In *Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems*, CHI '20, ACM, pp. LBW080:1–LBW080:7 (2020).
- [18] Richardson, M., Durasoff, M. and Wang, R.: Decoding Surface Touch Typing from Hand-Tracking; In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology*, ACM, pp. 686–696 (2020).
- [19] Dudley, J., Benko, H., Wigdor, D. and Kristensson, P. O.: Performance Envelopes of Virtual Keyboard Text Input Strategies in Virtual Reality; In *Proceedings of the 2019 IEEE International Symposium on Mixed and Augmented Reality*, ISMAR 2019, IEEE, pp. 289–300 (2019).
- [20] Speicher, M., Feit, A. M., Ziegler, P. and Krüger, A.: Selection-Based Text Entry in Virtual Reality; In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, CHI '18, ACM, pp. 647:1–647:13 (2018).
- [21] Boletsis, C. and Kongsvik, S.: Controller-based Text-input Techniques for Virtual Reality: An Empirical Comparison; In *International Journal of Virtual Reality*, (2019).
- [22] Yu, D., Fan, K., Zhang, H., Monteiro, D., Xu, W. and Liang, H.-N.: PizzaText: Text Entry for Virtual Reality Systems Using Dual Thumbsticks; In *IEEE Transactions on Visualization and Computer Graphics*, (2018).
- [23] Accot, J. and Zhai, S.: More Than Dotting the I's — Foundations for Crossing-Based Interfaces; In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '02, ACM, pp. 73–80 (2002).
- [24] Tu, H., Huang, S., Yuan, J., Ren, X. and Tian, F.: Crossing-Based Selection with Virtual Reality Head-Mounted Displays; In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, CHI '19, ACM, pp. 618:1–618:14 (2019).
- [25] Apitz, G. and Guimbretière, F.: CrossY: A Crossing-Based Drawing Application; In *Proceedings of the 17th Annual ACM Symposium on User Interface Software and Technology*, UIST '04, ACM, pp. 3–12 (2004).
- [26] Bi, X., Chelba, C., Ouyang, T., Partridge, K. and Zhai, S.: Bimanual Gesture Keyboard; In *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology*, UIST '12, ACM, pp. 137–146 (2012).
- [27] iNFINITE: iNFINITE | VR Headset Database and Utility; (2021). <https://www.infinite.cz/projects/HMD-tester-virtual-reality-headset-database-utility>.
- [28] MacKenzie, I. S. and Soukoreff, R. W.: Phrase Sets for Evaluating Text Entry Techniques; In *CHI '03 Extended Abstracts on Human Factors in Computing Systems*, ACM, pp. 754–755 (2003).
- [29] American National Corpus Project: Open American National Corpus | Open Data for Language Research and Education; (2021). <http://www.anc.org/>.
- [30] Arif, A. S. and Stuerzlinger, W.: Analysis of Text Entry Performance Metrics; In *2009 IEEE Toronto International Conference Science and Technology for Humanity (TIC-STH)*, IEEE, pp. 100–105 (2009).
- [31] Bangor, A., Kortum, P. and Miller, J.: Determining What Individual SUS Scores Mean: Adding an Adjective Rating Scale; In *Journal of Usability Studies*, Vol. 4, No. 3, pp. 115–123 (2009).
- [32] Kabbash, P. and Buxton, W. A. S.: The “Prince” Technique: Fitts’ Law and Selection Using Area Cursors; In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '95, ACM Press/Addison-Wesley Publishing Co., pp. 273–279 (1995).