

筑波大学大学院博士課程

システム情報工学研究科修士論文

文字列選択およびコマンド実行のための
携帯情報端末向け片手操作手法

安藤 宗孝

修士（工学）

（コンピュータサイエンス専攻）

指導教員 志築 文太郎

2019年3月

概要

スマートフォンの片手操作時に文字列選択および選択した文字列を利用した操作（以降、文字列操作）を行うためには親指を画面上部に伸ばす必要があるが、親指の届く範囲は限られている。このため、これらの操作を自由に行う事は難しい。我々は、片手操作時に画面上部に親指を伸ばす必要なく、文字列操作が可能となる手法を開発した。我々の手法はソフトウェアキーボードのいずれかのキーを押しつつ特定の動作を行う事により、キャレット操作または文字列選択を行う。さらにキーを離した際に、押下していたキーの種類に応じたカット、コピーなどのコマンド操作を選択していた文字列に対して行う。我々の手法はソフトウェアキーボード部のみを用いて文字列選択が可能であるため、親指を上には伸ばす必要がない。さらに文字列操作を1タッチ内にて行う事が可能である。

目次

第1章	はじめに	1
1.1	スマートフォンにおける文字列操作	1
1.2	スマートフォンにおける文字列操作を片手にて行う際の問題点	3
1.3	本研究の目的とアプローチ	4
1.4	貢献	4
1.5	本論文の構成	5
第2章	関連研究	6
2.1	ターゲットサイズに関する研究および小さなターゲットに対する操作手法	6
2.2	指が届きにくい範囲に対する操作手法	7
2.3	キャレット操作, 文字列選択, および文字列操作手法	7
2.4	傾きを用いた操作手法	9
2.5	スライドを用いた操作操作	10
第3章	予備実験1: スマートフォンの傾け方の調査	11
3.1	実験協力者および実験機器	11
3.2	実験内容	11
3.3	実験結果および考察	12
第4章	傾け操作を用いた文字列操作手法	15
4.1	傾け手法の設計	15
4.1.1	操作手順	15
4.1.2	キャレット操作および文字列選択	15
4.1.3	コマンド操作	17
4.2	実装	17
4.2.1	傾きによる速度決定と位置決定の比較	18
4.2.2	キャレットおよび選択範囲の位置決定	18
第5章	傾け手法の性能評価	20
5.1	実験協力者および実験端末	20
5.2	実験内容	20
5.3	実験結果および考察	22
5.3.1	操作速度	22

5.3.2	ユーザビリティ	23
第 6 章	指のスライドを用いた文字列操作手法	26
6.1	設計	26
6.1.1	操作手順	26
6.1.2	キャレット操作および文字列選択	26
6.1.3	コマンド操作	28
6.2	実装	28
第 7 章	予備実験 2：指のスライド量に対するキャレット移動量の調査	29
7.1	実験設計	29
7.1.1	実験協力者	29
7.1.2	実験機器	29
7.1.3	実験内容	29
7.2	実験結果	30
7.2.1	分析	30
7.2.2	実験協力者のコメント	32
7.3	考察	35
第 8 章	手法比較実験	37
8.1	実験設計	37
8.1.1	実験協力者	37
8.1.2	実験機器	37
8.1.3	実験タスク	37
8.2	実験結果	39
8.2.1	操作速度	39
8.2.2	ユーザビリティ	40
System Usability Scale		40
コメント		41
8.3	考察	41
第 9 章	議論	45
9.1	文字による影響	45
9.1.1	文字の大きさ	45
9.1.2	文字幅	45
9.2	ユーザの状況および周囲の環境による影響	45
9.3	実装の改善	46
9.3.1	上下方向のキャレット操作および文字列選択	46
9.3.2	傾きを用いた文字列選択の改善	46

第 10 章 まとめ	47
謝辞	48
参考文献	49
付 録 A 実験の際に用いた各種書類	60
A.1 承諾書	61
A.2 実験同意書	62
A.3 ビデオ画像の公開についての承諾書・承諾変更書	65
A.4 実験に関する説明（傾け手法の性能評価実験）	66
A.5 実験に関する説明（手法比較実験）	68
A.6 実験前アンケート（傾け手法の性能評価実験）	71
A.7 実験前アンケート（手法比較実験）	72
A.8 System Usability Scale	73

目 次

1.1	既存手法におけるキャレット移動および文字列操作. 上:既存手法におけるキャレット操作. キャレットを移動させたい目的の位置を直接タップする事によってキャレットを移動させる. 下:既存手法における文字列操作.	2
1.2	スマートフォンの把持姿勢. a) タッチスクリーン下部に対してタッチ操作を行う際の把持姿勢. b) スマートフォン中央部に対してタッチ操作を行う際の把持姿勢. c) タッチスクリーン上部に対してタッチ操作を行う際の把持姿勢.	4
3.1	スマートフォンの傾け軸.	12
3.2	スマートフォンを左右方向に傾けた際のそれぞれの傾け軸における傾きの変化の関係.	14
4.1	傾け手法におけるキャレット移動および文字列操作. 上段:傾け手法を用いたキャレット操作. 下段:傾け手法を用いた文字列操作手法. 文字列 (“whatever”) をコピーしている.	16
5.1	傾け手法の性能評価実験における実験アプリケーション.	21
5.2	各セッションにおける全実験協力者の平均所要時間.	23
5.3	傾け手法を使用した際の, 各セッションにおける各実験協力者の所要時間.	24
5.4	各セッションにおける P1 を除く実験協力者の平均所要時間.	24
5.5	各セッションにおける全実験協力者の平均所要時間の 60 セッション目までの近似曲線.	25
5.6	各セッションにおける P1 を除く実験協力者の平均所要時間の 20 セッション目までの近似曲線.	25
6.1	スライド手法におけるキャレット移動および文字列操作. 上段:スライド手法におけるキャレット操作. ユーザは4段目のキーを押下後, タッチスクリーンから指を離すことなくスライドする事によってキャレットを移動させることができる. 下段:スライド手法における文字列操作. 目的の操作が割り当てられたキーを押下した後, タッチスクリーンから指を離すことなくスライドする事によって, スライドした方向に選択範囲が拡張される. さらに, 指を離すことによって最初に押下したキーに割り当てられたコマンドが実行される.	27

7.1	スライド操作の予備実験に使用したアプリケーション. ターゲットキーはスライド操作開始時に押下すべきキーを示している. キャレットの目的位置はハイライトされた2文字の間である. この場合, <i>space</i> および <i>i</i> がハイライトされているため, <i>i</i> の直前が目的位置となる.	31
7.2	100 試行あたりにかかった時間の平均. 誤差棒は標準偏差を表す.	33
7.3	ターゲットキーを探す時間を除く 100 試行あたりにかかった時間の平均. 誤差棒は標準偏差を表す.	33
7.4	100 試行あたりにcaretが目的位置を通り過ぎた回数の平均. 誤差棒は標準偏差を表す.	33
7.5	100 試行あたりのタッチ回数の平均. 誤差棒は標準偏差を表す.	34
7.6	100 試行あたりに指を離した際にcaret位置が目的位置から1文字分ずれていた回数の平均. 誤差棒は標準偏差を表す.	34
7.7	100 試行中において, 2回以上のタッチを要した試行数. 誤差棒は標準偏差を表す.	34
7.8	実験に使用したスマートフォン (Xperia XZ) のピクセル数.	36
8.1	手法比較実験にて使用したアプリケーション.	38
8.2	全実験協力者における各手法ごとの1セッションあたりにかかった時間の平均. 誤差棒は標準偏差を表す.	40
8.3	操作時間の実験結果. 上:それぞれの手法における1セッションあたりにかかった時間の平均. 下:上の図に近似曲線を追加し, 20セッションまで表示したもの. 誤差棒は標準偏差を表す.	42
8.4	それぞれの手法における実験協力者ごとの各セッションにかかった時間.	43
8.5	手法ごとの SUS スコア. 誤差棒は標準偏差を表す.	44

表目次

3.1	傾け調査の結果. 左右上下:基本姿勢と比較した際の傾きの変化量. 基本姿勢: 基本姿勢におけるスマートフォン把持角度.	13
4.1	コマンドのキーへの割り当て例.	17
8.1	手法比較実験にて得られたコメント.	41

第1章 はじめに

近年，スマートフォンと呼ばれるタッチスクリーンおよび様々なセンサを搭載した片手把持可能なサイズの情報端末が世界中で利用されている．スマートフォンを使用している際に発生する操作の一つとして，表示されている文字列を選択し（以降，文字列選択），コピー，カット，またはペーストなどのコマンドを実行する事がある（以降，文字列選択およびコマンドの実行を合わせて文字列操作と記す）．この操作は，例えば，意味の分からない単語をコピーし，WEBブラウザに貼り付け，検索する事によって意味を調べる操作の一部として使用される．我々は，本論文において，スマートフォンにおける現状の文字列操作に存在する問題点を示し，この問題点を解決する文字列操作手法を提案する．さらに提案した文字列操作手法の評価を行う．

本章において，まず，スマートフォンにおける，文字列に対する，コピー，カット，もしくは削除などの操作の現状を述べる．さらに，このスマートフォンにおける文字列操作の問題点を示す．次に，本研究における問題解決のためのアプローチを示す．また，本研究の貢献を記し，最後に本論文の構成を示す．

1.1 スマートフォンにおける文字列操作

代表的なスマートフォンのオペレーティングシステムである，iOS¹およびAndroid²において，ユーザは以下の手順にて文字列操作を行う事ができる．

1. 選択したい文字列を長押しすることにより，文字列選択状態に移行する．
2. 選択範囲変更用のアジャスタを押下し，スライドさせることにて移動させ，選択範囲を調節する．
3. タッチスクリーンから指を離した後，一定時間後に選択範囲上部に表示されるメニューにて目的の操作をタップし，コマンド操作を行う．

また，図 1.1 にこの手法（以降，既存手法）を用いて文字列をコピーする手順を示す．既存手法は，文字列選択およびコマンド操作を，選択したい文字列に対して直接タップする事により実行できるため，わかりやすいという利点がある．

¹<https://www.apple.com/ios>

²<https://www.android.com>

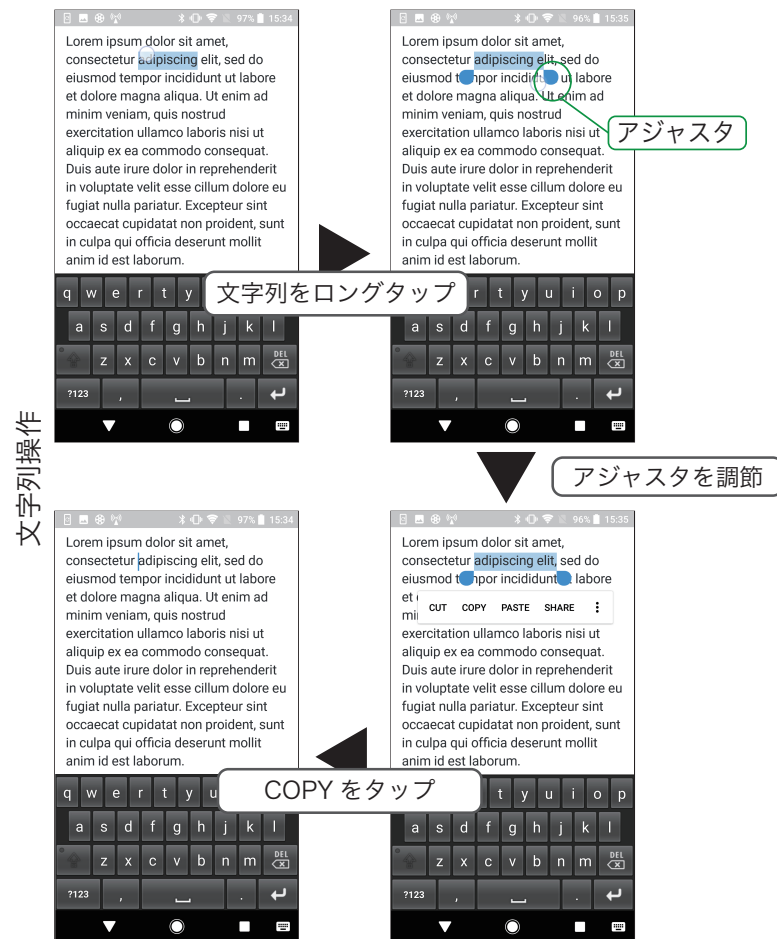


図 1.1: 既存手法におけるキャラクタ移動および文字列操作. 上: 既存手法におけるキャラクタ移動. キャラクタを移動させたい目的の位置を直接タップする事によってキャラクタを移動させる. 下: 既存手法における文字列操作.

1.2 スマートフォンにおける文字列操作を片手にて行う際の問題点

前節にて述べた既存手法を用いたスマートフォンにおける文字列操作には、3つの問題点がある。

- 文字列選択を開始するために、表示されている文字列を直接ロングタップするため、選択対象の文字列に指を伸ばす必要がある。この際、表示されている文字列に指が届かない場合、図 1.2 のように、ユーザはスマートフォンの持ち方（以降、把持姿勢）を変える必要がある [NM15]。把持姿勢の変更はユーザに煩わしさを感じさせると共に、変更時にスマートフォンを落下させる可能性がある。既存手法において、選択対象の文字列がタッチスクリーン下部に存在する場合は、容易にタッチ可能だが、タッチスクリーン上部に存在する場合には大きく指を伸ばす必要がある。特に、文字列を編集している際に文字列選択を実行しようとした場合、タッチスクリーン下部にはソフトウェアキーボードが表示されているため、選択対象文字列は必然的にタッチスクリーン上部に表示される事となる。このため、指を大きく伸ばす必要がある。また、ここ数年、スマートフォンに搭載されるタッチスクリーンが大型化してきているため、タッチスクリーンにおける指の届きにくい範囲が増えている [LMBH18]。これらの事から文字列選択を開始するために直接文字列をタッチする事はユーザに負担をかける動作となっている。
- 文字列を指にてタッチしようとした際、指が文字に対して大きいため、目的の文字を正確にタッチする事が難しいという問題がある（以降、Fat finger 問題 [SRC05]）。この問題は文字列に限らず、タッチスクリーンにおいて小さな対象をタッチ操作する際においても発生する。さらに、この問題は選択対象が密集している場合に発生しやすい。本研究にて焦点を当てている文字列選択は、選択対象となる文字がタッチスクリーン上における小さいオブジェクトであり、さらに文字である特性上密集しやすいため、この問題が発生しやすい。
- 指を伸ばした際に、選択対象の文字列が指にて隠れてしまい、正確なタッチができなくなる問題がある（以降、オクルージョンの問題）。本研究にて焦点を当てている文字列選択においては、選択対象となる文字が小さく、指にて覆い隠されやすいためこの問題も発生しやすい。

これらの問題に加えてコマンド操作時に発生する問題もある。既存手法においては、コマンド操作を行う際にメニューからコマンドを選択する。この際、メニューの表示項目数には限りがあるため、様々な操作を行うためには、目的の操作を選択するためにメニューを複数回選択し、目的の操作を探す必要がある。この操作は、複数回のタップを必要とするため、時間がかかる。

本研究においては、本節にて述べた既存手法の問題点を解決するために、複数の手法を提案、実装し、既存手法と比較する。

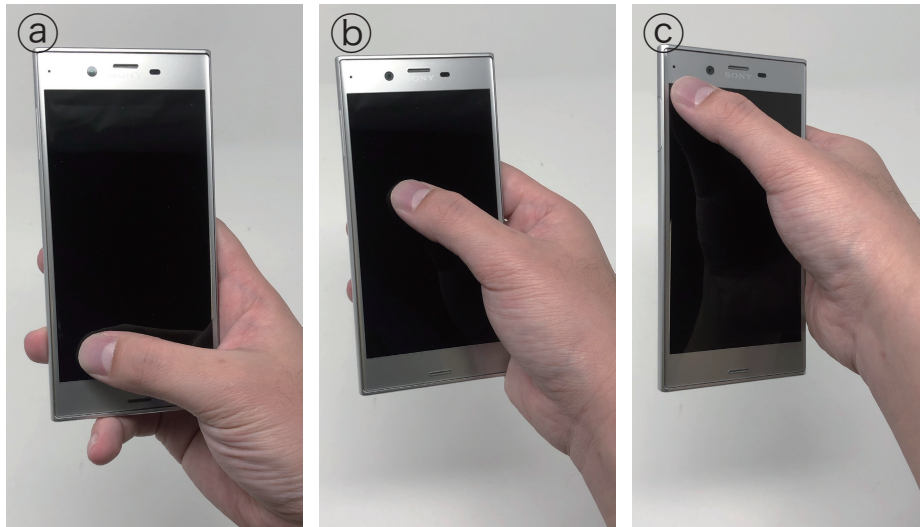


図 1.2: スマートフォンの把持姿勢. a) タッチスクリーン下部に対してタッチ操作を行う際の把持姿勢. b) スマートフォン中央部に対してタッチ操作を行う際の把持姿勢. c) タッチスクリーン上部に対してタッチ操作を行う際の把持姿勢.

1.3 本研究の目的とアプローチ

本研究の目的は前節にて述べた問題を解決することである. このために, 我々は本研究にて, 新たな文字列操作手法を提案し, 実装および評価を行う. 我々の提案する手法は, 指を上には伸ばすことなくコマンド操作を行うために, ソフトウェアキーボードを利用する. ソフトウェアキーボードの表示領域はタッチスクリーン下部であるため, 把持姿勢を変更せずとも, ソフトウェアキーボード内の多くの領域をタッチする事が可能である. また, ソフトウェアキーボードに配置されている多くのキーを利用して多彩なコマンド操作を行う事ができる. さらに, キートップの文字に関連したコマンド操作をそれぞれのキーに割り当てる事により, 多くのコマンド操作が存在する場合においても, 個々のキーに割り当てられた操作を想起し, 適切なコマンドを実行できる.

また, このソフトウェアキーボードを利用したコマンド操作と組み合わせて利用するための, 文字列選択手法も併せて示す. 本論文においては, 2種類の文字列選択手法を示す. 一方は, スマートフォンを傾ける操作を用いた手法 (以降, 傾け手法) である. もう一方は, スマートフォンのタッチスクリーン上にて指をスライドさせる操作を用いる手法 (以降, スライド手法) である. 本論文においてはこれらの手法の詳細を述べるとともに, これらの手法と既存手法を比較した結果も述べる.

1.4 貢献

本論文の貢献は以下の 5 点である.

- スマートフォンの傾きを用いた文字列操作手法
- タッチスクリーン上にて指をスライドさせる操作を用いた文字列操作手法
- ユーザがスマートフォンを傾ける際、どのように傾けるかの調査結果
- 傾け手法およびスライド手法の設計および実装
- 既存手法、傾け手法、およびスライド手法の比較実験結果

1.5 本論文の構成

第1章においては本論文の背景、研究対象および目的を述べる。第2章にて、関連研究に対する本研究の位置づけを述べる。第3章から第5章にて、提案手法の内の1つである、傾け操作を用いた手法を詳述する。続く第6章から第7章にて、傾け操作と異なる提案手法として指のスライドを用いた手法を述べる。第8章にて、既存手法、傾け手法、およびスライド手法を対象とした比較実験を述べる。第9章にて、本研究に関連する議論を示し、第10章にて本研究のまとめを述べる。

第2章 関連研究

これまでに、スマートフォンにおけるターゲット選択に関する調査、および小さいターゲットに対する操作手法、片手持持時に指が届きにくい範囲に対する操作手法の研究が多数なされている。また、我々の研究と同じく、キャレット操作、文字列選択、および文字列操作手法も多数提案されている。さらに、傾きを用いた操作手法、指のスライドを用いた操作手法もこれまでに研究されている。本章にて、これらの研究の概要と関連研究に対する本研究の位置づけを述べる。

2.1 ターゲットサイズに関する研究および小さなターゲットに対する操作手法

Fat finger 問題 [SRC05] は Siek らが提唱した、タッチスクリーンにおけるターゲットサイズおよび指の大きさに関連する問題である。ユーザがタッチスクリーンにてターゲットをタッチしようとする際、特に指がターゲットよりも大きい場合において、指の大きさ故に意図したターゲットを正確にタッチする事ができない事がある。これは、ターゲットが密集している場合、ターゲットではなく、他のオブジェクトを選択しやすくなるため、より発生しやすい。この問題は我々の手法において、解決すべき問題の1つである。タッチスクリーンが小さい端末におけるターゲットサイズの研究も行われている [PKB06]。この研究においては、片手持持された端末においてターゲットを親指を用いた直接タッチにて選択する場合には、ターゲットサイズを 9.6 mm 以上にすることにより、パフォーマンスおよび操作感を損ないにくいとの結果が示されている。我々が操作の対象とする文字はこの研究にて示されているターゲットサイズの要件よりも小さいため、直接タッチを用いる手法においてはパフォーマンスを損なう可能性がある。このため、我々の手法は直接タッチではなく、間接的な操作を用いて文字列操作を実現する。

ターゲットサイズが小さい場合に正確にターゲットをタッチする手法がある。Shift [VB07, GZMI09] は、タッチした部分をタッチスクリーンのタッチ位置ではない部分に虫眼鏡の様に表示する。これによって、ユーザはタッチしている部分を見ることが可能であるため、fat finger 問題の発生を防ぐことができる。我々の手法もこの手法と同様にこの節にて述べた問題に対する手法である。一方、我々の手法はより文字列操作を行いやすいように設計されている。

2.2 指が届きにくい範囲に対する操作手法

スマートフォンは、片手把持可能なサイズの情報端末である。しかし、片手にて把持し、把持した手によって操作を行おうとする際、指が届かない範囲が存在する。例えば、右手にてスマートフォンを把持した際には、親指はスマートフォンの右下に配置されるため、端末の上部および左部には指が届きにくくなる。このようなスマートフォンの指が届かない、または届きにくい範囲を調査した研究が存在する [BLO14]。また、この指が届きにくい範囲に対する操作を可能とする手法がこれまでに数多く提案されている。RapidSelection は、スマートフォンのタッチスクリーン全体をタッチするのではなく、タッチスクリーン上に矩形を表示し、矩形内のタッチをタッチスクリーン全体に転送することによって、指の届きにくい範囲をタッチできるようにした手法である [YHHH13]。Karlson らは、タッチスクリーン全体を縮小表示したタッチパッドをタッチスクリーン上に表示し、これをタッチする事によってタッチスクリーン全体を操作する事ができるようにした [KB08]。TouchShield は広い面積のタッチを行った際に、タッチ位置に物理キーに対応したウィジェットを表示させることによって、指の届かない位置に存在するこれらのキーへの操作を実現する手法である [HL13]。Kim らは、広い面積のタッチまたはベゼルスワイプ [RT09] を行った後、ドラッグ操作を行う事により、ユーザは画面領域全体を指の方向に移動させることによって指の届く位置にターゲットを移動させる事により、指が届かなかった位置に対するタッチ操作を可能とする手法を提案した [KYL12]。TouchOver [大西 14] は画面下部におけるタッチイベントを画面上部へと転送することによって、指が届かない画面上部に対する間接操作を可能とする。なお、スマートフォンを振るという操作によって画面下部への直接操作と画面上部への間接操作を切り替える。MagStick [RHL08] は親指の動きと逆方向に移動するカーソルによって、指の届きにくい画面上部に対する操作を可能とした。iOS には画面の上部にある項目を画面の下半分に移動し、画面上部の項目に対するタッチを行いやすくする簡易アクセス機能がある [Inc18a]。

これらの手法および機能は我々の手法と同じく、指が届きにくい範囲における操作を容易にする。しかし、文字列選択や文字列操作は文字入力に付随して行われることがあり、この場合、これらの手法においては頻繁に通常の操作との切り替えが発生する。これに対して我々の手法は文字入力中に使用する際、通常操作と共存可能であり、操作の切り替えおよび画面の切り替えなどは発生しない。

2.3 キャレット操作、文字列選択、および文字列操作手法

キャレット操作および文字列選択のための手法がこれまでに提案されてきている。Scheibel ら [SPM⁺13] はタッチスクリーン上に表示したジョイスティックに基づくキャレット操作手法を示した。ユーザにテキスト編集を行わせる実験からフォントサイズが大きい場合には既存手法よりも遅い事が示されたが、フォントサイズが小さい場合には既存手法と同等の速度にてキャレット操作を行う事が可能であった。Fix and Slide [SOSS16] はドラッグによってキャレットを移動するのではなく、画面上のキャレットの位置を固定した状態にて、文字列全体をドラッグによって移動する事にてキャレットの相対位置を移動させる事により、キャレットの

位置を移動させる手法である。iPhoneには圧力を測定する事ができるタッチスクリーン（以降、感圧タッチスクリーン [Inc15]）上にて、ソフトウェアキーボードを強くタッチしドラッグすることによってキャレット移動を行う事ができる機能がある。また、この機能はタッチする圧力を変えることによりキャレット移動に伴う文字列選択も可能である [Inc18b]。これに加えて、Gogueyら [GMG18]は感圧タッチスクリーンを用いたキャレット移動や文字列選択の問題点を報告し、拡大鏡を始めとする視覚的フィードバックを実装する事によってより使いやすくすることを示している。Push-Push [HL15]は感圧タッチスクリーンを用いた文字列選択の手法である。強くスクリーンを押下した点から選択状態となり、次に強くスクリーンを押下するまでにキャレットが移動した範囲を選択する。ForceEdge [AMC17]は感圧タッチスクリーンまたは感圧タッチパッドを持つ情報端末向けの手法であり、タッチ圧力を変化させることによってスクロールしながらの文字列選択を容易にした手法である。

これらの手法は我々の手法と同じく、キャレット操作または文字列選択が可能な手法である。我々の手法は、傾きを用いて文字列選択を行う点、および文字列選択後のコマンド操作を容易にする点がこれらの手法と異なる。

これまでに述べた、キャレット操作および文字列選択手法は、スマートフォンにおける手法である。一方、ラップトップまたはデスクトップコンピュータにおける手法も提案されている。例えば、Push-Edge and Slide-Edge [MAQ⁺15]はタッチパッドを利用したコンピュータにおける文字列選択手法である。コンピュータでの文字列選択において、文字列選択をしつつスクロール操作を行うするためには、選択状態にてマウスカーソルをウインドウ下部にて止めスクロールが実行される事を待つ必要があった。この手法においては、親指にてタッチパッドをタッチしたまま、人差し指にてタッチパッド右側をスライドする事によって選択範囲を変更する事ができる。

さらに、文字列選択後のコマンド操作を容易にする手法も研究されている。Gestures and Widgets [FIM13]はソフトウェアキーボード上にて左右スワイプ、円を描くなどのジェスチャを行う事によってキャレットの移動、文字列選択、またはコマンド操作を行う手法である。この手法は我々の手法と同じく、指を伸ばす必要なく文字列操作を行う事ができる。この手法においてはジェスチャを用いて文字列操作を行うが、我々の手法はソフトウェアキーボードのキーに対応したコマンドを実行するため、キートップの文字からコマンドを想起することができる利点を持つ。PalmTouch [LKB⁺18]は画面上部をタッチしようとする際に手のひらがタッチスクリーンに触れることから、親指または手のひらによるタッチを識別し、この組み合わせを用いて操作を拡張する手法である。この手法の使用例の1つとして、手のひらの位置を起点としてコマンド操作に必要なメニューを親指の届きやすい範囲に表示した。Eadyらはスマートフォンサイズの携帯型情報端末を横向きに持った際に、キャレット操作および文字列操作を行う手法として、画面の右上および左上を折り曲げる動作を利用した [EG15]。Eadyらは、プロトタイプとして曲げセンサを内蔵したシリコンにてスマートフォンサイズの板を作製し、この板に対する折り曲げ操作を認識した。TouchTap [FM17]は複数本の指を用いて行うマルチタッチジェスチャを用いてキャレットの移動および文字列操作を行う研究である。BezelCopy [CPZO14]はベゼルスワイプ [RT09]を用いてコマンド操作を行う手法である。こ

これらの手法は我々の手法と同じく、文字列操作をより容易にする手法である。しかし、直接タッチする手法であるため、オクルージョンおよび Fat finger 問題の発生を抑えることはできない。

2.4 傾きを用いた操作手法

我々の提案する手法の内の1つである傾け手法と同じく、把持した端末の傾きを用いて操作を行う手法はこれまでに数多く研究されている。Rekimoto [Rek96] は端末自体を傾ける事を端末への操作に使用する手法を示した。この手法は傾きとボタンを組み合わせる事によって縦型および円形のメニュー操作を可能とする。KeyTilt [HSM10] はソフトウェアキーボードにおいて、1つのキーに5つの文字を割り当て、キーをタッチしつつ端末を傾ける事によって5つの文字から1つを選択する手法である。この手法では、8つのキーを用いてアルファベットおよび特殊文字を入力できるため、キー1つあたりを大きく表示することが可能である。TiltType [PCS+02] は手首に装着可能な端末において、8方向に配置した文字それぞれの方向に傾ける事によって文字入力を行う。また、ユーザは端末の上下に配置されたボタンを押すことによって配置されている文字を入れ替えることが可能である。Yeo ら [YPC+17] はソフトウェアキーボード上の文字を指にてなぞって入力するジェスチャキーボードに傾け操作を適応した手法を提案した。この手法は、キーボード上に傾きに応じて移動するポインタを表示し、指を用いてキーをなぞる動きを傾きによるポインタの移動に置き換えた入力手法である。TiltText [WB03] はフィーチャーフォンにおける文字入力の際、傾け操作を用いる事によってボタンを押す回数を減らす手法である。Tsandilas ら [TABB14] はスマートフォンの傾きに応じて、タッチスクリーン上の表示をスマートフォンの傾きと同じ向きに傾ける事により、画面上の指から遠い位置に対するタッチを容易にする手法を示している。Tilt to Scroll [OO05] は携帯情報端末の画面操作において、携帯情報端末自体を傾ける事によって画面をスクロールする手法である。Crossan ら [CMBMS09] は、加速度センサを付けた帽子を使用する事によって、装着者の頭の角度を取得し、頭の傾けをスマートフォンの操作に用いた。InclineType [GV15] はスマートウォッチにおいて、傾け操作を利用した事によって文字入力を行う手法である。傾きを用いて入力したい文字を選択する。Hinckley ら [HPSH00] は手のひらサイズの大きさのコンピュータに対して加速度センサを含む複数のセンサを取り付け、端末を持ち上げる、握るなどの動作を検知する事により、画面を点灯させる、アプリケーションを起動するなどの機能を提案している。この中の1例として、傾きの変化に基づいて画面をスクロールする手法も提案されている。Cho ら [CMSK07] は携帯情報端末における、画像を見るアプリケーションにおいて、端末の傾きに基づいてスクロールし、さらに、傾きの大きさから一度に表示する画像数を変化させる手法を示した。Tilt & Touch [DRPL11] は傾け操作を3次元映像を見る際の視点操作に用いる手法である。Chang ら [CLKS15] は大画面スマートフォンにおいて、傾け操作を用いる事によって、指が届かない範囲を操作するために、画面の移動および縮小などを行う事ができる。CanalSense [AKST17, AKST18] は、センサを埋め込んだイヤホンを用いて、頭を傾ける動作を認識する。使用例として、認識した動作をスマートフォ

ンに対する入力に利用した。

提案手法はこれらの手法と同じく、傾きを用いて操作を行う手法であるが、提案手法は文字列操作を目的としている点がこれらの手法と異なる。

さらに傾け操作とタッチ操作以外の操作を組み合わせた手法も提案されている。Blowatch [Che15] はスマートウォッチにおいて傾け操作と息を吹きかける動作を組み合わせた手法である。ユーザは息を吹きかけた後に傾け操作を行う事によって、ズームインまたはズームアウト、およびスクロールを行う。

また、傾きを用いた操作手法の性能評価も行われている。Rahman ら [RGIS09] はフィーチャーフォンを用いて傾け可能な範囲およびその範囲における傾け操作の精度を調査し、調査結果を用いて傾け操作のための設計指針を示した。Fitchett ら [FC09] は iPod touch を用いた実験を行い、傾け操作を用いたスクロールとフリックを用いたスクロールを比較した。この結果、座位において傾け操作を用いたスクロールの方がフリックを用いたスクロールよりも素早く操作可能であった。Teather ら [TM14] はタブレット端末における傾け操作において、傾きからカーソルの画面上の位置を決定する実装、および傾きからカーソルの移動速度を決定する実装の比較を行った。この結果、傾きから画面上の位置を決定する実装がもう一方の実装と比較して約2倍のパフォーマンスを示した。Shima ら [SOT+16] はスマートウォッチにおける傾け操作の精度調査を行い、調査結果からスマートウォッチにおける傾け操作の設計指針を示した。Gilbertson ら [GCCV08] は3Dドライブゲームにおいて、キー入力による操作と傾けによる操作を比較した。結果として、ユーザは傾けによる操作の方が楽しいと感じたと報告された。Sad ら [SP09] は傾け操作がフィッツの法則に従うかどうかの検証を行った。これはセンサを取り付けたPDAを使用したポインティングタスクおよびスクロールタスクにて行われ、傾け操作を用いたポインティングおよびスクロールはフィッツの法則に従うという結果を得た。

これらの性能評価を参考に、傾けを用いた手法の実装を行う。

2.5 スライドを用いた操作操作

我々の提案する文字列操作手法の内の1つであるスライド手法と同じく、スライド操作を用いて文字列選択を行う手法がこれまでに提案されている。LeavingSelect [島14, 島17] は、2本指にてタッチスクリーンをタッチした後、指を1本タッチした状態にて、もう一方の指を離し、残した指をスライドする事により、スライドした範囲を矩形選択する手法である。また、これに加えて、矩形選択後にもう一方の指をタッチする事によってカットもしくはコピーなどの操作を行う事ができる。Mac OS には、タッチパッドを3本指にてスライドする事によって矩形選択を行う事ができる機能がある [Inc17]。

我々はこれらの手法と同様に指をスライドする事によって、文字列選択を行う手法を示す。

第3章 予備実験1：スマートフォンの傾け方の調査

傾け手法を実現するためには、ユーザがスマートフォンを傾ける際に、スマートフォンの回転軸のうち、どの軸が傾けた際に変化するかを調査する必要がある。これは、手の回転軸およびスマートフォンの回転軸に差があり、ユーザが手のある回転軸に沿って動かしたとしても、スマートフォン側においては複数の回転軸に変化が現れている可能性が考えられるためである。また、傾け手法においてはユーザはタッチスクリーンの内容を視認する必要およびソフトウェアキーボードのキーを押下している必要があるため、これらの要素も傾けた際の角度変化に影響を及ぼすと考えられる。このため、これら複数の条件を考慮したスマートフォンを傾けた際の角度変化の調査実験を行った。本実験の目的はスマートフォンを左方向、右方向、上方向、および下方向（以降、左右上下方向）に傾けた際の *Roll*, *Pitch*, および *Yaw* (図 3.1) の変化を調査する事である。本実験においてはスマートフォンを片手持持した状態にて、左右上下方向に傾けた際のスマートフォンの角度を計測した。本実験にて計測した角度を傾け手法の実装に利用する。

3.1 実験協力者および実験機器

大学生および大学院生のボランティア 12 名（男性 9 名，女性 3 名，平均年齢 23.7 歳）を実験協力者とした。すべての実験協力者が日常的にスマートフォンを使用していた。また、すべての実験協力者が普段スマートフォンを右手にて把持していた。実験開始時に実験協力者の手長を計測した。この結果、平均 18.4 cm（最長 20.1 cm，最短 16.3 cm）であった。

実験にて使用するスマートフォンとして SONY 製の Xperia XZ（端末サイズ：高さ 146 mm × 幅 72 mm × 厚さ 8.1 mm，重さ 161 g，スクリーンサイズ：5.2 インチ，OS：Android 8.0.0）を用いた。

3.2 実験内容

実験協力者には普段スマートフォンを持つ持ち方および普段タッチスクリーンを見ている角度にて、スマートフォンを右手を用いて把持してもらった（以降、基本姿勢）。その後、基本姿勢から左方向、右方向、上方向、および下方向の順に傾けてもらった。なお、それぞれの方向にスマートフォンを傾ける前に基本姿勢に戻すよう指示した。実験協力者間にて左右ま

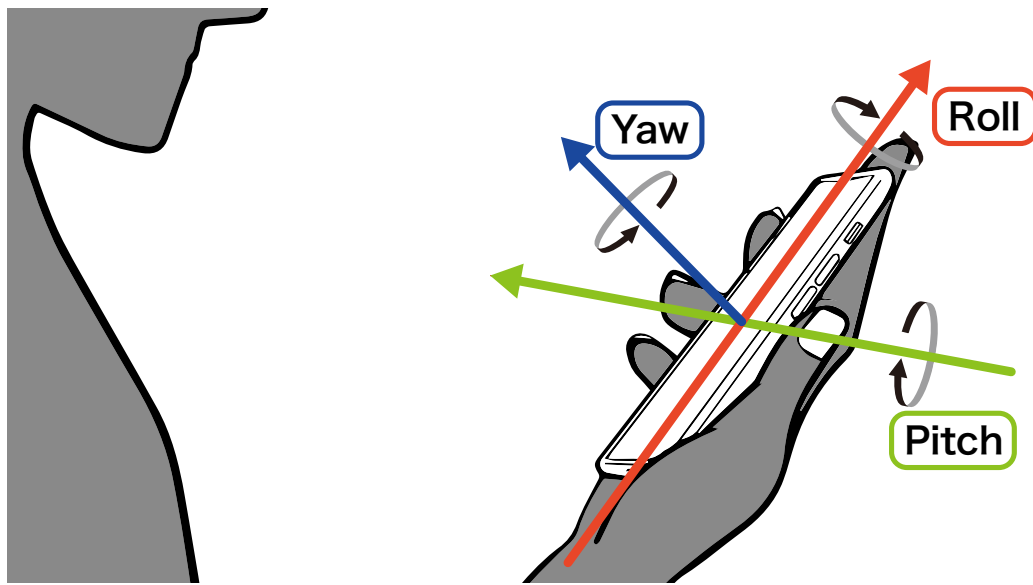


図 3.1: スマートフォンの傾け軸.

たは上下が逆になる事を防ぐために、事前に実験協力者に4方向の傾け方向を指定した。本実験におけるスマートフォンを傾ける範囲の指示として、実験協力者にはスマートフォンを傾けた際にタッチスクリーン上のキャラットの位置および文字列が無理なく確認できる範囲において傾けるように指示した。これは、我々の手法は、ユーザがスマートフォンのタッチスクリーンを見つつ操作を行う事を想定しており、キャラットおよび文字列を視認不可能な範囲までスマートフォンを傾けた際の角度は想定と異なるが故に、実装に利用できないためである。傾け範囲の指示と同様に、本実験は我々の手法に利用する事を目的としているため、実験協力者には、*m* キーを押下した状態にてスマートフォンを傾けた後、前述した傾け範囲内にて最も傾けた状態で指を離すよう指示した。指を離した際の角度を傾けた際の最大の角度として記録した。すなわち指を画面に押下した状態における傾き角度を取得した。本予備実験にて *m* キーを使用した理由は、ソフトウェアキーボードにおいてアルファベットが割り当てられているキーのうち、最も右下に位置するキーだったためである。左方向、右方向、上方向、下方向の順にそれぞれ傾ける事を1セッションとし、1人あたり5セッション行うよう指示した。傾け1回を1試行とすると、合計240試行(4方向×5セッション×12人)行った。なお、本予備実験は椅子に座った状態にて行ってもらった。

3.3 実験結果および考察

実験結果を表 3.1 に示す。今回は左右上下方向の中から、左右方向に傾けた場合に注目する。これは本論文における傾け手法は、水平方向に関する実装のみを行ったためである。表 3.1 から、左右に傾けた場合において、*Roll* だけでなく、*Pitch* および *Yaw* が変化している

表 3.1: 傾け調査の結果. 左右上下: 基本姿勢と比較した際の傾きの変化量. 基本姿勢: 基本姿勢におけるスマートフォン把持角度.

傾け方向	<i>Roll</i>	<i>Pitch</i>	<i>Yaw</i>
左	-61.34°	-24.80°	25.30°
右	56.78°	-22.19°	15.35°
上	121.85°	2.39°	34.06°
下	-10.33°	-31.81°	9.53°
基本姿勢	13.03°	67.60°	-37.44°

事が分かる. さらに, *Roll* のみが傾け角度の正負が逆転しており, *Pitch* および *Yaw* においては傾け角度の正負は同一である事が分かった. また, 左右に傾けた際, 左方向に傾ける方がすべての回転角において変化量が大きくなる事が分かった. これは, すべての実験協力者が右手把持であったことが原因であると考えられる. 画面上の文字およびキャラクタの位置が無理なく確認できる範囲にて傾ける事を指示していたため, 右手把持の場合, 右手と頭の位置関係から左側に傾けた場合の方が右側に傾けた場合よりも画面を確認できる範囲が広がった可能性がある. このため, スマートフォンの操作に主に左手を使う実験協力者を対象とした調査を行った場合, 異なる結果が得られると考えられる.

図 3.2 にそれぞれの実験協力者が左右方向にスマートフォンを傾けた際に, スマートフォンの傾きがそれぞれの回転角ごとにどのように変化しているかを示す. P12 を除く全ての実験協力者において, 左右にスマートフォンを傾けた際, ほぼ同じようにスマートフォンの角度が変化している. P12 の実験結果は他の実験協力者と比べてより広範囲に変化している. 図 3.2 左上の角度分布から, P12 の結果は, 中央に集まる点群および両脇に集まる点群の 2 つに分かれているように観察される. 中央の点群は他の実験協力者と大きく差はないと考えられるが, 両脇の点群は他の実験協力者と比べて, 大きく異なる動きをしている. この理由に関して, P12 は全ての実験協力者の中にて, 最も手長が短い (16.3 cm), つまり最も手の小さい実験協力者であったため, 傾けた際にスマートフォンを安定して把持することができず, 思わぬ角度に変化した可能性がある. すなわち, 先に述べた中央の点群は, この実験協力者がスマートフォンを安定して把持することができていた範囲であり, 両脇の点群は安定を欠いた把持によって角度が変化してしまった際の角度である可能性がある. ただし, 手の小さい人, または P12 に特有の傾け方の癖である可能性もある. したがって, この実験に関して, より正確な結果を得るためにはさまざまな手の大きさの人に対して実験を実施する必要があるが, 本論文における傾け手法の実装には本予備実験にて得られた結果を用いる.

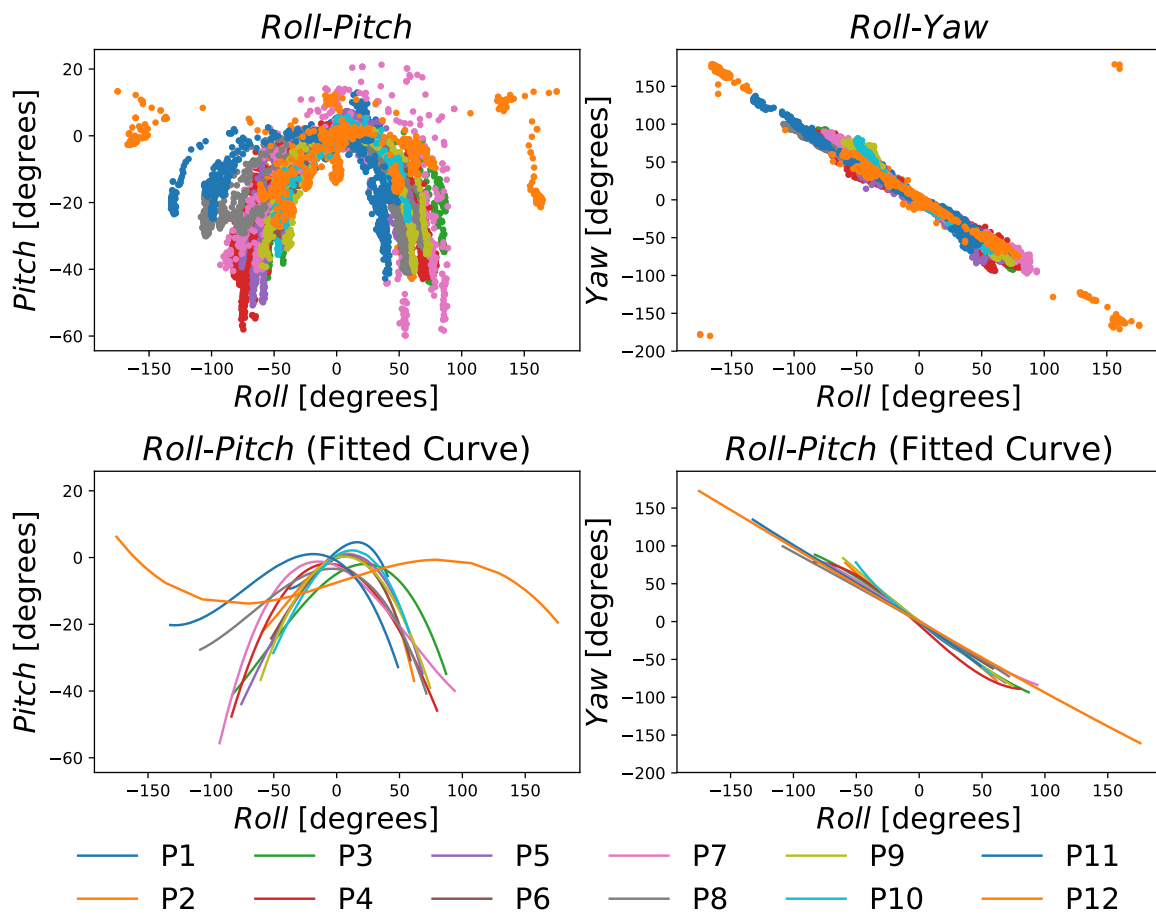


図 3.2: スマートフォンを左右方向に傾けた際のそれぞれの傾け軸における傾きの変化の関係.

第4章 傾け操作を用いた文字列操作手法

傾け角度の調査実験にて得た左右方向への傾け時の角度を用いて、傾きを用いたキャレット操作および文字列選択の設計および実装を行う。

4.1 傾け手法の設計

我々の提案する文字列操作手法の1つである傾け手法の設計を述べる。

4.1.1 操作手順

傾け手法におけるキャレット操作および文字列操作の手順を図4.1に示す。傾け手法における文字列操作は以下の手順にて実行できる。

1. 目的のコマンドが割り当てられているキーを押す。
2. キーを押したままスマートフォンを傾けると、傾けた方向に選択範囲が拡張される。
3. 指を離すと、離した際に選択されていた文字列に対してコマンド操作が行われる。

この操作は、画面下部に表示されているソフトウェアキーボードを用いて操作を行うため、スマートフォンを片手把持している際にも指を上には伸ばさずに操作を行う事が可能である。また、この操作は、複数回のタッチを必要とせず、1回のタッチにて文字列操作を行う事ができる。なお、1文字も選択していない状態にて指を離した場合、コマンド操作は実行されず、擬似的にキャンセルする事ができる。

4.1.2 キャレット操作および文字列選択

傾け手法を用いてキャレット操作を行うためには、ソフトウェアキーボードの *m* キーを押した状態にてスマートフォンを傾ける。この際のスマートフォンの傾きに応じてキャレットが移動する。*m* キーをキャレット操作を行うためのキーとした理由は、アルファベットを入力するキーとして最も右下に存在する事、すなわち、右手把持した際に親指に近い位置に存在する事、および *m* キーから英単語“move”を連想する事ができる事からである。

文字列選択を行うためには、ソフトウェアキーボードの *m* キー以外のキーを押した状態にてスマートフォンを傾ける。スマートフォンの傾きに応じて選択範囲が変化する。キャレット操作および文字列選択は、現状横方向にのみ実行可能である。

キャレット移動



文字列操作



図 4.1: 傾け手法におけるキャレット移動および文字列操作. 上段: 傾け手法を用いたキャレット操作. 下段: 傾け手法を用いた文字列操作手法. 文字列 (“whatever”) をコピーしている.

表 4.1: コマンドのキーへの割り当て例.

キー	コマンド
<i>c</i>	コピー
<i>v</i>	選択範囲を上書きしてペースト
<i>Shift</i>	大文字と小文字を変換
<i>b</i>	文字列をブラウザにて検索
<i>s</i>	文章内を選択した文字列で検索
<i>l</i>	文字列を用いたアプリケーションの起動
<i>m</i>	キャレット操作

4.1.3 コマンド操作

コマンドは文字列選択時に押下したキーを離すことによって実行できる。キーを離した際に選択していた文字列を用いてコマンドを実行する。なお、傾けた後にキーを離した場合は、文字列操作が行われるため、通常の文字入力が行われない。通常の文字入力を行うためには、キーを押下した後、スマートフォンを傾けずに指を離す。1文字以上選択されなかった場合には、通常の文字入力が行われる。押下していたキーによって異なるコマンドを実行することができる。

表 4.1 にキーへのコマンド操作の割り当て例を示す。表 4.1 に示すように様々なコマンド操作を割り当てることが可能である。また、*m* キーを用いたキャレット操作は *m* キーを押した際のコマンド操作として捉える事が可能であり、手法としての一貫線を持たせている。ソフトウェアキーボードのキーそれぞれにコマンドを割り当てることが可能であるため、PC におけるキーボードショートカットの様に多くの機能を実現できる。また、キートップの文字に関連するコマンドを割り当てることにより、キートップから割り当てられたコマンドを想起することが可能である。

コンピュータにて用いられるコピー、ペーストなどの基本的なコマンドに加えて、選択している文字列を使用したコマンドも実行可能である。例えば、表 4.1 において *b* キーに割り当てられている「文字列をブラウザにて検索」では、スマートフォン側にて設定されたブラウザを用いて選択している文字列の検索を行う。*l* キーに割り当てられている、「文字列を用いたアプリケーションの起動」とは、選択されている文字列に関連するアプリケーションを起動するものである（例、選択された文字列: `wifi`, 起動するアプリケーション: `wifi` の設定）。このような文字列を用いたアプリケーションの起動を用いる事によってソフトウェアキーボードをアプリケーションランチャの様に使用する事が可能である。

4.2 実装

傾け手法の実装を述べる。

4.2.1 傾きによる速度決定と位置決定の比較

傾きを用いてキャラクタ操作および選択範囲の変更を行う際、2種類のキャラクタ位置および選択範囲の決定方法が主に利用されている。1つは、傾け操作をキャラクタ移動速度や選択範囲の変更量の決定に用いる手法（以降、移動速度決定手法）である。もう1つは、傾け操作をキャラクタや選択範囲の位置決定（以降、位置決定手法）に用いる手法である。これまでに、これらの比較を行った研究が複数存在する。Theatherら [TM14] はタブレット端末におけるポインティングタスクを用いてこれらの手法を比較している。結果として、位置決定手法が移動速度決定手法の約2倍の操作速度であった。Oakleyら [OO05] は手のひらサイズのコンピュータにおけるメニュー操作タスクを用いてこれらの手法を比較した。結果、位置決定手法の方が移動速度決定手法に比べて素早く操作可能である事を示した。これらの小型端末を用いた先行研究における実験の結果に基づき、我々の手法においては位置決定手法を用いる。

4.2.2 キャラクタおよび選択範囲の位置決定

傾け角度の調査実験の結果を考慮してキャラクタ位置および選択範囲の決定手法を実装する。位置決定手法の1つとして、三角関数を用いてタッチスクリーン上に表示したポイントの位置決定を行う手法が先行研究にて示されている [SOT+16]。この手法では傾け角度 θ に対応するポイントの位置 (PP) を式 (4.1) を用いて求める。

$$PP = \sin \theta \times cd_ratio \quad (4.1)$$

cd_ratio は定数であり、スマートフォンの傾け量に対するディスプレイ上のポイント移動距離の比である。しかし、この方法を用いてキャラクタ位置を決定した場合、一般的に使用されているプロポーションアルファベットにおいては、アルファベットの文字幅が一定でない事から選択しやすい文字と選択しにくい文字が発生する。文字幅が広い文字は、選択に大きな傾けを必要とする一方、文字幅が狭い文字は小さな傾けにて選択される。このとき、文字幅が狭い文字は広い文字よりも正確に選択できる角度が小さくなるため、正確に選択する事が難しくなる。このため、今回の実装においては、1文字あたりの傾け量が等しくなるように実装する必要がある。また、傾け角度の調査実験から、左右において傾ける事ができる角度の大きさが異なる事が分かったため、これも考慮して実装する。

このため、今回の実装は次の手順にてキャラクタ位置を決定する。

1. 左右どちらへ移動するかは $Roll$ を用いて決定する。
2. 左右ごとにキャラクタ位置を決定する。

まず、左右の決定においては、左右方向に傾けた際 $Roll$ のみが左右対称的に変化することが分かっているため、これを利用する。つまり、 $Roll$ の値が正負のどちらに変化したかを用い

てキャレット位置の変化方向を決める。次にキャレット位置は、式 (4.2) を用いて求める。

$$CP = \sqrt{\left(\frac{nR}{mR}\right)^2 + \left(\frac{nP}{mP}\right)^2 + \left(\frac{nY}{mY}\right)^2} \times cc_ratio \quad (4.2)$$

nR , nP , および nY はそれぞれ、現在の *Roll*, *Pitch*, および *Yaw* であり, mR , mP , mY はそれぞれ傾き角度の調査実験にて得たそれぞれの最大角度である。 cc_ratio は定数であり, キャレットの移動文字数に対する傾け量の比である。今回の実装においては経験的に $cc_ratio = 20$ とした。

第5章 傾け手法の性能評価

傾け手法と既存手法における文字列選択および文字列操作にかかる時間を比較する実験を行った。本実験において我々は、キャレット操作、文字列選択、および文字列操作を含めた一連の操作にかかる時間の比較を行った。また、System Usability Scale (SUS) を利用し、既存手法と傾け手法のユーザビリティを比較した。

5.1 実験協力者および実験端末

実験協力者として大学院生6名（男性4名，女性2名，平均年齢23.3歳，P1-P6）を雇用した。このうち3章にて述べた傾け角度の調査を行った予備実験に参加していたのは4名（P1-P3, P5）であった。すべての実験協力者が日常的にスマートフォンを使用していた。また，すべての実験協力者が普段スマートフォンを右手にて把持していた。実験開始時に実験協力者の手長を計測した。この結果，平均18.3 cm（最長19.2 cm，最短16.3 cm）であった。実験にて使用するスマートフォンは3章における予備実験1と同じくXperia XZを用いた。実験協力者には，報酬として1640円支払った。

5.2 実験内容

実験協力者には着座姿勢にてこの実験を行ってもらった。実験協力者には2種類の手法を用いて文字列操作を行うよう指示した。図5.1に実験において使用したアプリケーションの画面を示す。図5.1における文字列Aが選択対象文字列である。文字列Aは文字列Bの部分文字列となっており，実験協力者は文字列B中の青地にて示された文字列Aの部分に対して指示された手法を使用して文字列操作を行う。以下に，2種類の文字列操作手法の実験における操作内容を示す。

傾け手法 実験協力者は，まず，文字列選択の開始点を変更するために，選択対象文字列の端にキャレットを移動させる。これは m キーを押下しながらスマートフォンを傾ける事により行う。次に，文字列の選択を行う。これは m キー以外のキーを押下しながらスマートフォンを傾ける事により行う。傾け角度を調整することによって選択範囲を調整する。最後に，文字列を選択した状態にて指を離すことによって，選択した文字列に対して文字列操作が行われる。

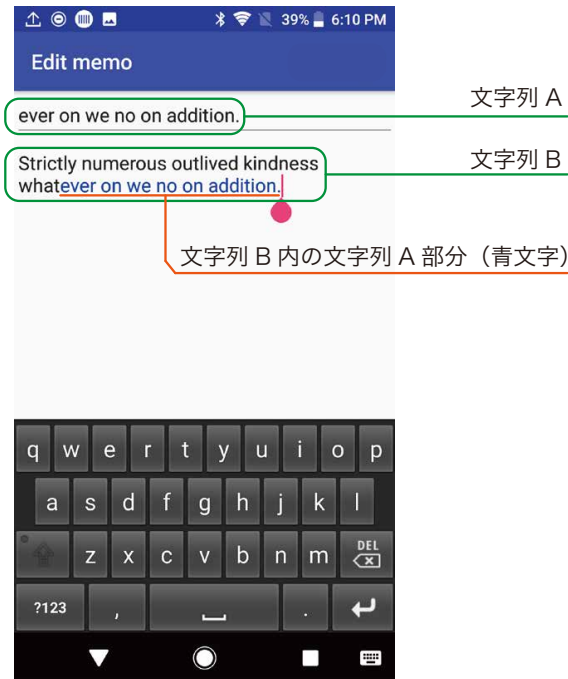


図 5.1: 傾け手法の性能評価実験における実験アプリケーション.

既存手法 実験協力者は、まず、文字列を直接ロングタップする事によって文字列を選択状態にする。この状態にて、選択範囲変更用のアジャスタをドラッグし選択範囲を変更する。次に、文字列を選択した状態にて指を離すことにより、選択した文字列の上部にメニューが表示される。メニューの中から *Next* をタップする事によって選択した文字列に対して文字列操作が行われる。

文字列操作を行った際に、選択対象文字列に対して正しく文字列操作が行われた場合には、文字列が更新され、次の文字列が表示される。選択されている文字列が選択対象文字列と異なっている場合には、文字列は更新されない。文字列を正しく 1 回選択する事を 1 試行とし、11 試行を 1 セッションとする。セッションは実験協力者が最初の文字列に対し、文字列操作を正しく行った直後から開始される。つまり、各セッション第 2-11 試行の 10 試行におけるデータを本実験にて取得した。実験協力者には手法ごとに連続して 10 セッション行ってもらった。よって、本実験にて取得したデータは各実験協力者につき合計 200 試行 (= 10 試行 × 10 セッション × 2 手法) である。また、手法間の順序効果を打ち消すために、実験協力者を 2 つのグループに分け、片方のグループは傾け手法を先に行い、もう片方のグループは既存手法を先に行った。疲労による影響を軽減するために、実験協力者にはセッション間に 1 分以上の休憩を取ってもらった。それぞれの手法の使用感を調査するために、各手法の全セッション終了後に、実験協力者には SUS に回答してもらった。SUS はシステムのユーザビリティを計る手法であり、10 個の質問に答えることにて 0 から 100 点のスコアが示される。スコアは高い方が良く、スコアの平均値は 68 になるように定義されている。また、自由記述式のアンケート

トにも回答してもらった。本実験は1人あたり約75分の時間を要した。実験に使用した文字列はランダムに生成されたものであり¹、1文あたり平均8.8単語である。また、文字列更新後における選択対象文字列の探索時間を減らすために、文字列B中の文字列Aは文字色を青色に設定した。なお、既存手法においては自然言語処理を用いた機能により単語の端に選択範囲が吸着するが、設定の変更は行わず、既定の仕様にて使用している。さらに、既存手法においては、表示されたメニューにてNextではないコマンドを選択する事を防ぐために、メニューに表示されるコマンドをNextのみとした。

5.3 実験結果および考察

本実験において得られたデータを用いて、操作速度およびユーザビリティの観点から既存手法および傾け手法の比較評価を行う。

5.3.1 操作速度

図5.2に各手法ごとの全実験協力者が各セッションにかかった時間の平均を示す。また、併せて手法ごとの近似曲線を示す。10セッションのデータを用いて対応のあるt検定を行った場合、既存手法および傾け手法間に有意な差が存在した ($p = .00 < .05$)。これは、既存手法の方がセッションを完了するためにかかった時間が短かったことを示す。今回の実験においては、練習セッションなしに本番セッションを行っているため、セッション数を重ねるごとに操作時間が短くなっている。特に、傾け手法においては、セッション数に応じて操作時間が短くなっている。このため、セッション数の経過による影響を検証するため、10セッションを5分割し、それぞれにおいて傾け手法と既存手法を比較した。結果として、1-2セッションは有意差あり ($p = .016 < .050$)、3-4セッションは有意差無し ($p = .072 > .050$)、5-6セッションは有意差あり ($p = .022 < .050$)、7-8セッションは有意差無し ($p = .215 > .050$)、9-10セッションは有意差無し ($p = .072 > .050$) となった。この結果から、ばらつきはあるものの、セッション数を重ねるごとに傾け手法と既存手法との差が減少する可能性がある。傾け手法および既存手法の操作時間の変化を調査するため、セッション数を増やした際の近似曲線を図5.5に示す。図5.5から、傾け手法および既存手法の近似曲線が56セッション目にて交差することが分かった。このため、本実験の結果においては傾け手法は既存手法よりも操作速度が遅かったものの、ユーザが傾け手法に慣れる事により、この差が減少する可能性が示された。

また、実験協力者それぞれの各セッションにかかった時間を図5.3に示す。図5.3からP1が各セッションを完了するのににかかった時間が、不安定に変化しており、他の実験協力者と比べても時間かかっていることが分かる。これは、3章における予備実験にて、他の実験協力者と異なる傾け方を示したP12が本実験におけるP1と同一人物であり、本実験にて用いた傾け手法の実装が予備実験の全実験協力者のデータを用いているため、P1の傾け方に合致して

¹<http://randomtextgenerator.com>

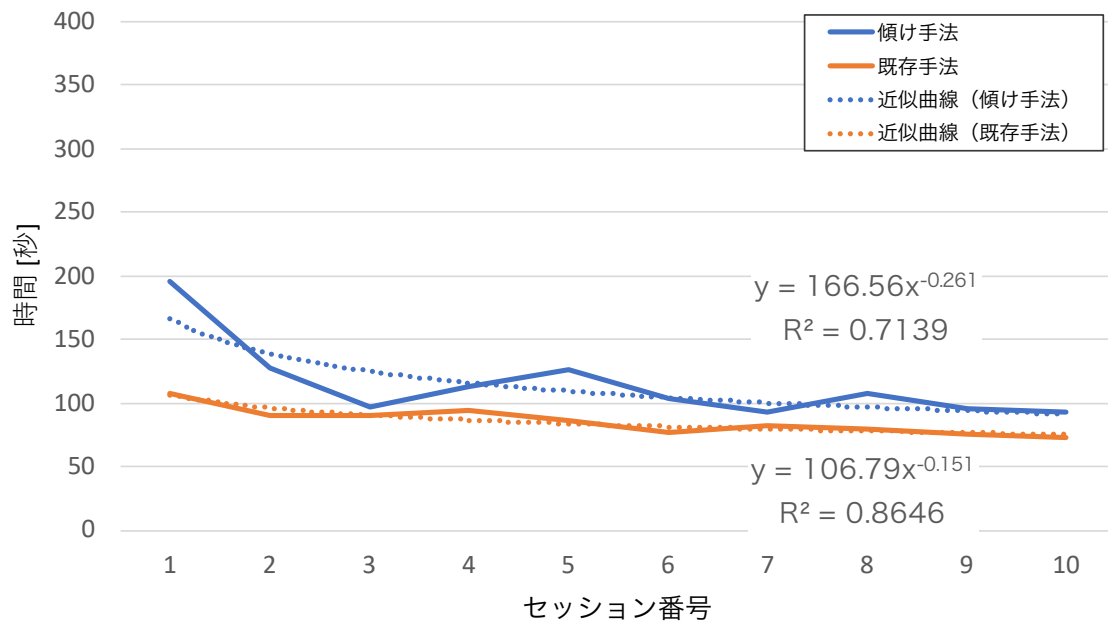


図 5.2: 各セッションにおける全実験協力者の平均所要時間。

いなかった可能性がある。このため、P1 を除く実験協力者のデータにおける平均試行時間の調査を行った (図 5.4)。結果として、P1 を含めた場合に比べ、より既存手法との差が小さくなった。また、図 5.5 に示す様に、P1 を除いたデータにおいては 13 セッション目において近似曲線が交差することが分かった。この結果から、より個人に合った実装を採用することによってより高い性能を実現する事ができる可能性がある。

5.3.2 ユーザビリティ

各手法終了後に実験協力者に回答してもらった SUS のスコアは、傾け手法が平均 61.7、既存手法が平均 58.8 であった。手法間に有意な差は見られなかった。また、SUS のスコアは平均値が 68 になるように定義されているため、傾け手法および既存手法ともに定義された平均値よりも低い結果となった。このため、両手法ともにユーザビリティの観点において改善が必要である。

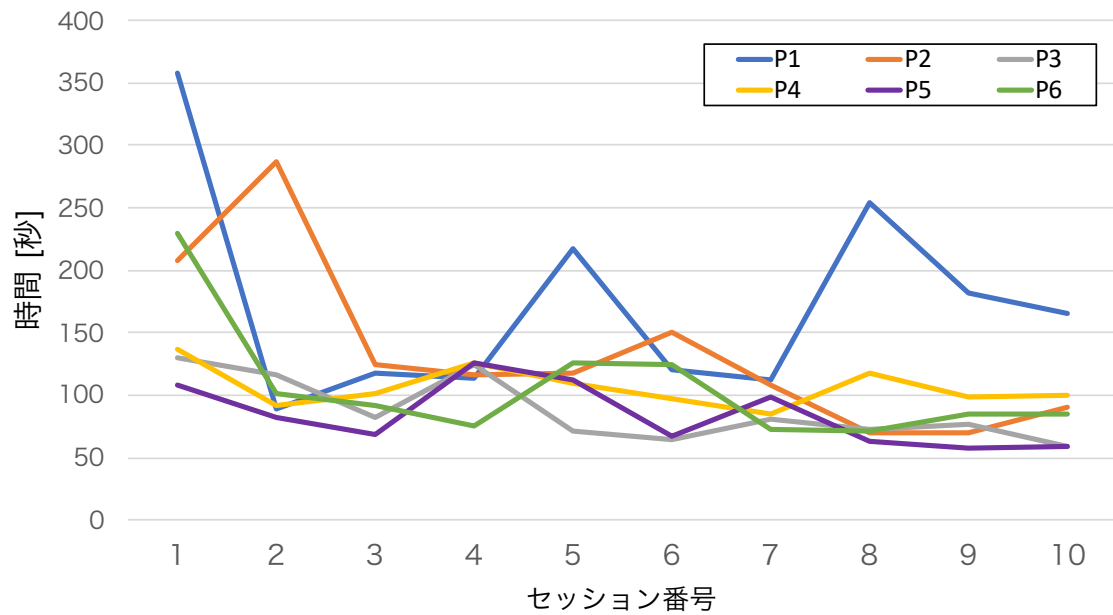


図 5.3: 傾け手法を使用した際の、各セッションにおける各実験協力者の所要時間。

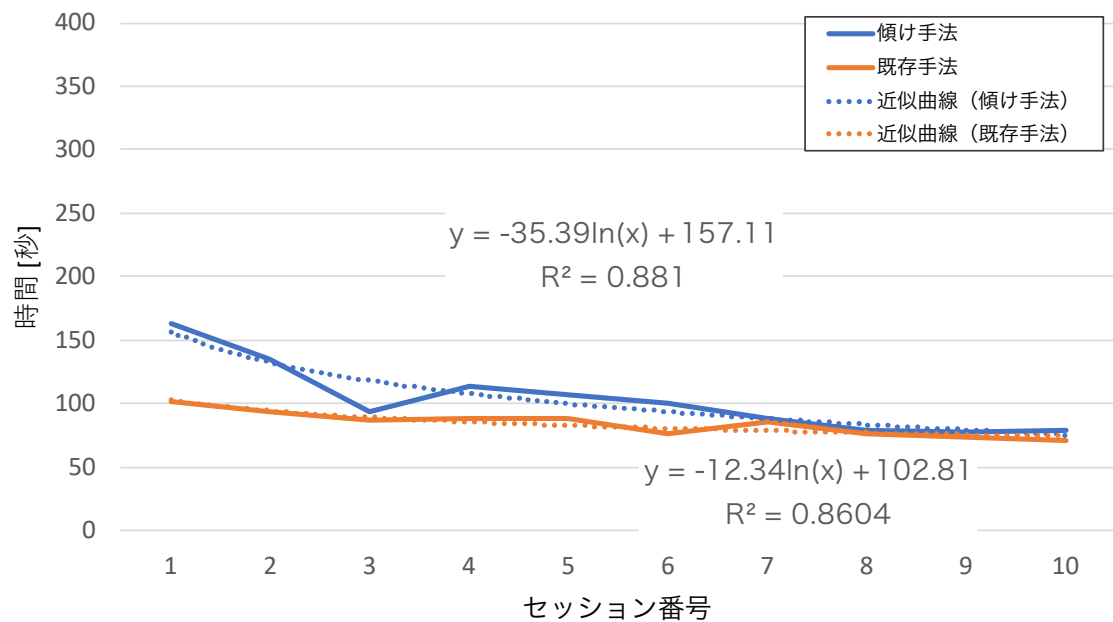


図 5.4: 各セッションにおける P1 を除く実験協力者の平均所要時間。

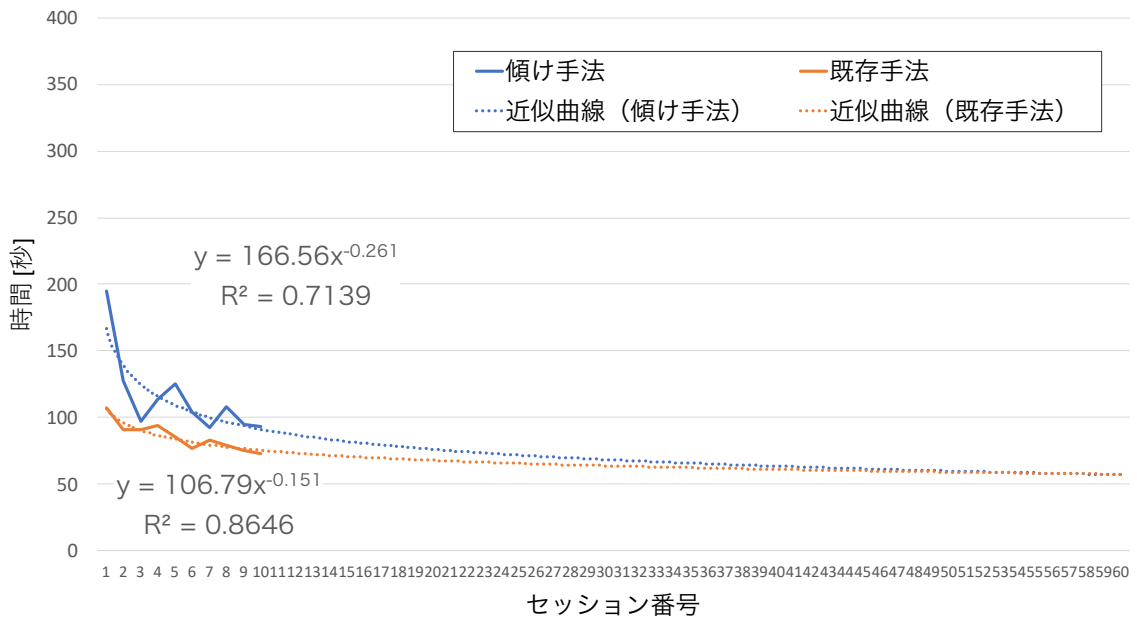


図 5.5: 各セッションにおける全実験協力者の平均所要時間の 60セッション目までの近似曲線.

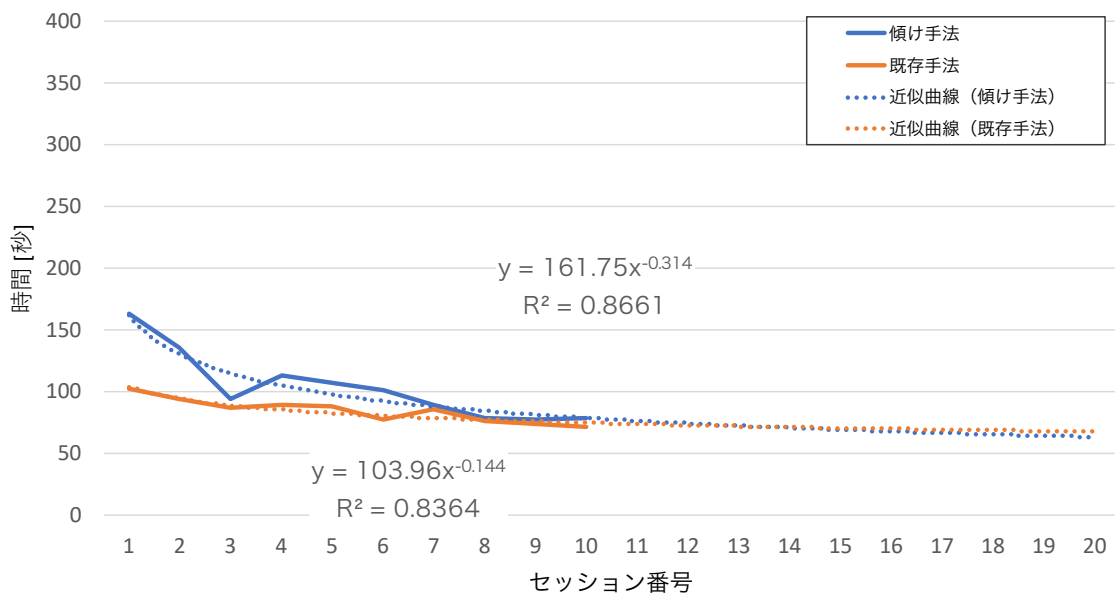


図 5.6: 各セッションにおける P1 を除く実験協力者の平均所要時間の 20セッション目までの近似曲線.

第6章 指のスライドを用いた文字列操作手法

5章にて行った実験の結果から、傾け方の個人差に起因する、傾け手法に対する得手不得手が存在した。このため、我々は、日常的に利用されており、個人差の影響が少ないと考えられるタッチスクリーンのみを使用した手法を提案する。本章にて述べるスライド手法は、タッチスクリーン上における指のスライドを用いる文字列操作手法である。本章においては、スライド手法の設計および実装を示す。

6.1 設計

我々の提案する手法の1つであるスライド手法の設計を述べる。

6.1.1 操作手順

スライド手法においては、キーを押下し、タッチスクリーン上にて指をスライドする動作を文字列操作に用いる。図6.1にスライド手法を用いたキャレット移動および文字列操作の例を示す。ユーザはキャレット操作および文字列選択を以下の手順にて行う。

1. 目的のコマンドが割り当てられているソフトウェアキーボードのキーを押下する。
2. タッチスクリーンから指を離すことなく、選択範囲を広げたい方向に指をスライドする。
3. 指を離すことによって選択範囲に対してコマンド操作を実行する。

この操作手順は傾け手法と同じく、画面下部に対するタッチのみにて操作を完了する事が可能であるため、指を伸ばさずに操作を行う事ができる。また、傾け手法と同様に1タッチにて文字列操作を行う事が可能である。

6.1.2 キャレット操作および文字列選択

スライド手法を用いてキャレット操作を行うためには、ソフトウェアキーボードの4段目のキーを押下した後、指をタッチスクリーンから離さずにスライドする。この際に指をスライドした向きに応じてキャレットが移動する。キャレット操作に割り当てられているキーが傾け手法手法と異なるのは、7章における予備実験にて、実験協力者からキーの押しやすさについてのコメントがあったためである。文字列選択を行うためにはソフトウェアキーボー

キャレット移動



文字列操作



図 6.1: スライド手法におけるキャレット移動および文字列操作. 上段: スライド手法におけるキャレット操作. ユーザは4段目のキーを押下後, タッチスクリーンから指を離すことなくスライドする事によってキャレットを移動させることができる. 下段: スライド手法における文字列操作. 目的の操作が割り当てられたキーを押下した後, タッチスクリーンから指を離すことなくスライドする事によって, スライドした方向に選択範囲が拡張される. さらに, 指を離すことによって最初に押下したキーに割り当てられたコマンドが実行される.

ドの4段目以外のキーを押下した後、指をスライドする。キャレット操作および文字列選択は傾け手法と同じく、現状横方向のみに実行可能である。なお、通常の文字入力を行うためには、入力したいキーを押した後、指をスライドせずにタッチスクリーンから指を離す事によって文字を入力する事ができる。また、スライドを開始した後、文字列操作をキャンセルしたい場合は、文字列を選択していない状態にて指を離すことによってキャンセルする事ができる。なお、この場合には文字入力が行われない。

6.1.3 コマンド操作

文字列選択後、指を離すことによって、選択した文字列を用いてコマンド操作を行う。スライド手法における文字列操作も、4章にて述べたコマンド操作と同じ利点がある。

6.2 実装

ユーザが最初にタッチスクリーンをタッチした際のタッチ座標（以降、初期座標）およびその後の指のタッチ座標（以降、現在座標）を比較する事によってキャレットまたは文字列の選択範囲を決定する。キャレットの移動量および文字列の選択範囲の変更量は同じ手法を用いて算出するため、代表してキャレット移動量の算出方法を示す。傾け手法と同じく、本手法においても1文字あたりに必要なタッチ座標の変化量を文字幅に依らず一定にする事により、文字幅が異なる文字であっても一定の感覚にて操作する事ができる。今回の実装においては式(6.1)を用いて初期座標および現在座標からキャレットの移動量(ACM^1)を求める。

$$ACM = (ITP - CTP) \times cd_ratio \quad (6.1)$$

ITP^2 は初期座標、 CTP^3 は現在座標である。なお、 ITP および CTP の単位はピクセルであり、 ACM の単位は文字である。例として、 ACM が-3の場合、初期キャレット位置から左方向に3文字移動する。 cd_ratio は、指のスライド量に対するキャレットの移動量の比を表している。 cd_ratio が大きいほど、少ないスライド量にて、大きくキャレットを移動する事ができる、または、大きな範囲を選択する事ができるが、キャレットが指の動きに対して敏感に反応するため、キャレットを目的の位置に止める、または、目的の範囲を正確に選択する事が困難になる。このため、 cd_ratio には適切な値を設定する必要がある。我々は cd_ratio の最適値を調査するために、予備実験を行った。

なお、本実装においては、選択が開始される基準は ACM が1以上または-1以下である事である。すなわち、指を10ピクセル以上移動させた時点から選択が開始される。このため、既存手法のような、長押しすることによって選択が開始される手法に比べ、より高速に選択を開始できる。

¹Amount of Caret Movement の略

²Initial Touch Position の略

³Current Touch Position の略

第7章 予備実験2：指のスライド量に対する キャレット移動量の調査

本章において、指のスライド量に対するキャレット移動量に関する予備実験について述べる。本予備実験の目的は、スライド手法を実装するために、指のスライド量に対するキャレットの移動量（以降、スライドCD比¹）を求める事である。スライドCD比を求めるため、実験協力者には、スライドCD比を3段階に分け、キャレットを目的の位置に移動させるタスクを行ってもらった。なお、本予備実験にて取得したスライドCD比はスライド手法において文字列選択の選択範囲時にも用いる。

7.1 実験設計

本予備実験に参加した実験協力者、実験機器、および実験タスクを述べる。

7.1.1 実験協力者

実験協力者は所属研究室の学生を含む、大学生および大学院生12名（男性11名、女性1名、平均年齢22.8歳）であった。実験協力者には、実験開始前に、普段左右どちらの手を用いてスマートフォンを操作するかを質問し、普段スマートフォンを使用する方の手にて実験タスクを行ってもらった。普段スマートフォンを使用する際に、左右どちらかに特定せず、両方の手を使用すると回答した実験協力者には、今回の実験においては右手にて実験タスクを行ってもらった。この結果、本予備実験においては全ての実験協力者が右手を使って実験タスクを行った。

7.1.2 実験機器

実験には、傾け角度の調査と同じく、スマートフォンとしてXperia XZを用いた。

7.1.3 実験内容

実験協力者は、3種類のスライドCD比を変更した条件（以降、スライドCD比条件）において、キャレットを目的の位置に移動させるタスクを行った。スライドCD比条件を、著者

¹スライド Control-Display 比の略

の経験から、0.05, 0.1, および0.2とした。また、キーの位置によって押しにくさ、およびスライドの行いやすさが変わることが考えられるため、押下するキーの指定を行った。

図 7.1 に実験に使用したアプリケーションを示す。実験協力者は図 7.1 に示すターゲットキーに表示されているキーを画面下部のソフトウェアキーボードにて最初に押下し、その後、指をスライドする事によって指示された位置（以降、目的位置）までキャレットを移動させて指を離す行為を実験タスクとして行った。目的位置は、図 7.1 の様にハイライトされた 2 文字の間である。目的位置は文字列 **B** の中からランダムにて設定されている。指を離した際にキャレットの位置が目的位置にあった場合、ターゲットキーおよび文字列が更新され、次のターゲットキーおよび文字列が表示される。指を離した際のキャレットの位置が目的位置と異なる場合、文字列は更新されない。この場合、実験協力者は再度ターゲットキーを押下し、キャレット移動を行う。また、指のスライド中に指が画面端まで到達し、キャレット移動が行えない場合には、指を離し、再度ターゲットキーを押下した後、指をスライドさせる。

ターゲットキー押下後、目的位置までキャレットを移動させ、指を離すことを 1 試行として、各実験協力者は 1 条件当たり 100 試行を行った。結果として、3600 試行（12 人 × 3 条件 × 100 試行）を行った。また、実験中に実験協力者からそれぞれの CD 比に対するコメントを取得した

100 試行中に特定のターゲットキーのみが多数指定されると実験結果に影響する可能性があるため、ターゲットキーの出現頻度を調整した。本予備実験にて使用したソフトウェアキーボードには 33 種類のキーが配置されているため、100 試行の内に 33 種類のキーが 1 種類当たり 3 回および *space* キーが 1 回ターゲットキーとして指定されるように設計した。ターゲットキーの出現順序はランダムに設定した。実験協力者毎の順序効果をなくすために、スライド CD 比条件毎にカウンターバランスを取った。この実験における、キャレットの初期位置は文字列 **B** の中からランダムにて設定した。この際、キャレットの初期位置および目的位置が同じとなった場合には、初期位置および目的位置が同一ではなくなるまで初期位置の再設定を行った。また、1 試行に対し、2 回以上のタッチが必要とする場合においては、2 回目のタッチにおいてもターゲットキーを押してキャレットを移動させるよう指示した。

7.2 実験結果

本予備実験にて得られたデータを用いて、各スライド CD 比における操作時間、タッチの回数などをスライド CD 比ごとに比較した。

7.2.1 分析

実験結果を図 7.2–7.7 に示す。図 7.2 に示される通り、100 回あたりの試行時間は、スライド CD 比が 0.2 条件の場合に最も長くなり、0.05 条件の場合に最も短くなった。正規性の検定を行ったところ、取得したデータに正規性がなかったため、ノンパラメトリック検定として Friedman の検定を行った。結果、スライド CD 比間に主効果が存在する事が分かった



図 7.1: スライド操作の予備実験に使用したアプリケーション. ターゲットキーはスライド操作開始時に押下すべきキーを示している. キャラットの目的位置はハイライトされた 2 文字の間である. この場合, *space* および *i* がハイライトされているため, *i* の直前が目的位置となる.

($p = 0.001 < 0.05$) . さらに, **Dunn** らの手法 [**Dun64**] を用いて多重比較を行ったところ, 0.05–0.2 条件間 ($p = 0.003 < 0.05$) および 0.1–0.2 条件間 ($p = 0.002 < 0.05$) に有意な差が存在する事が分かった.

図 7.3 は前段落にて述べた, 100 試行あたりの試行時間から, 各試行ごとの画面をタッチするまでの時間, つまりターゲットキーを探している時間を除いたものである. 取得したデータに正規性がなかったため, ノンパラメトリック検定の **Friedman** 検定を行った. 結果, スライド CD 比間に主効果が存在する事が分かった ($p = 0.000 < 0.05$) . さらに, **Dunn** らの手法を用いて多重比較を行ったところ, 0.05–0.2 条件間 ($p = 0.001 < 0.05$) および 0.1–0.2 条件間 ($p = 0.001 < 0.05$) に有意な差が存在する事が分かった.

図 7.4 のオーバーシュートとは, キャレット移動時にキャレットが目的位置を過ぎることを表している. 100 試行あたりのオーバーシュートの回数は, 0.2 条件の場合に最も多くなり, 0.05 条件の場合に最も少なくなった. 正規性の検定を行ったところ, 取得したデータに正規性がなかったため, ノンパラメトリック検定の **Friedman** 検定を行った. 結果, スライド CD 比間に主効果が存在する事が分かった ($p = 0.00 < 0.05$) . さらに, **Dunn** らの手法を用いて多重比較を行ったところ, 0.05–0.2 条件間 ($p = 0.00 < 0.05$) および 0.1–0.2 条件間 ($p = 0.007 < 0.05$) に有意な差が存在する事が分かった.

図 7.5 は 100 試行あたりのタッチの回数を示している. これは 0.5 条件の場合に最も多くなり, 0.1 条件の場合に最も少なくなった. 取得したデータに正規性があったため, 反復測定分散分析を行ったところ 1 セッションあたりのタッチ回数間に主効果は存在しなかった ($p = 0.10 > 0.05$) .

図 7.6 は 100 試行中において, 目的位置の 1 文字隣にて指を離した回数を示している. つまり, 指を上げる際にキャレットがずれてしまった, 1 文字移動量が足りなかったなどの要因によって 1 文字ずれてしまった回数を表す. 1 文字ずれてしまう事が多発する場合にはイライラするとのコメントを実験協力者から得ているため, ユーザビリティの点において重要である. 正規性の検定を行ったところ, 取得したデータに正規性がなかったため, ノンパラメトリック検定の **Friedman** 検定を行った. 結果, スライド CD 比間に主効果が存在する事が分かった ($p = 0.00 < 0.05$) . さらに, **Dunn** らの手法を用いて多重比較を行ったところ, 0.05–0.2 条件間 ($p = 0.00 < 0.05$) および 0.1–0.2 条件間 ($p = 0.02 < 0.05$) に有意な差が存在する事が分かった.

図 7.7 は, 100 試行中において, 1 試行に 2 回以上のタッチを必要とした試行数である. 取得したデータに正規性があったため, 反復測定分散分析を行ったところ, 100 試行あたりの 2 タッチ以上を必要とした試行数に有意差があった ($p = 0.006 < 0.05$) . さらに, **Bonferroni** 法を用いて多重比較を行ったところ, 0.05–0.1 条件間に有意な差が存在した ($p = 0.005 < 0.05$) .

7.2.2 実験協力者のコメント

実験中に実験協力者から複数のコメントを得た. 0.05 条件において, 「(キャレットを) 動かす距離が長い場合にめっちゃイライラする」, 0.2 条件において, 「1 文字ズレてしまいづらい」といったコメントがあった. また, 全条件, 特に 0.2 条件に, 指を離す際にキャレット位置が

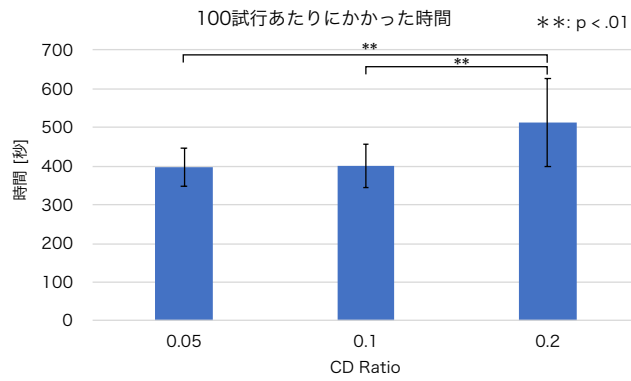


図 7.2: 100 試行あたりにかかった時間の平均. 誤差棒は標準偏差を表す.

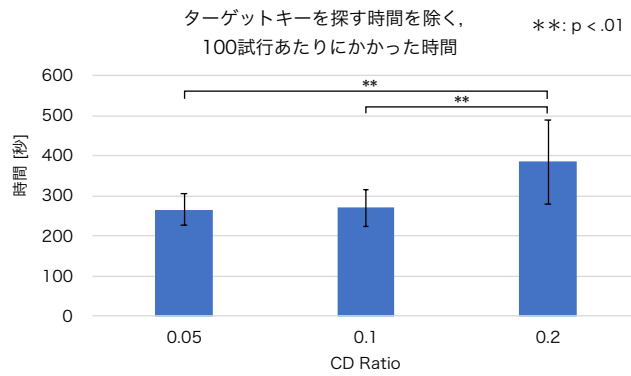


図 7.3: ターゲットキーを探す時間を除く 100 試行あたりにかかった時間の平均. 誤差棒は標準偏差を表す.

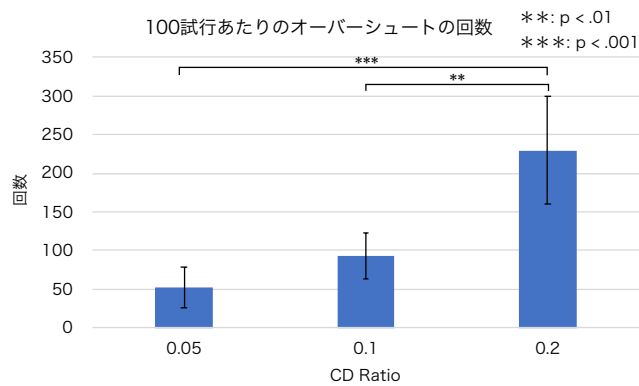


図 7.4: 100 試行あたりにキャレットが目的位置を通り過ぎた回数の平均. 誤差棒は標準偏差を表す.

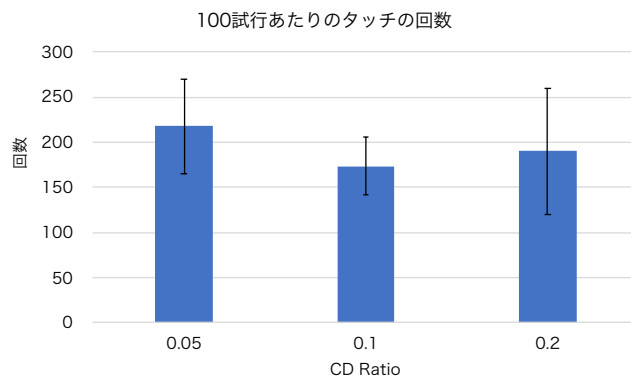


図 7.5: 100 試行あたりのタッチ回数の平均. 誤差棒は標準偏差を表す.

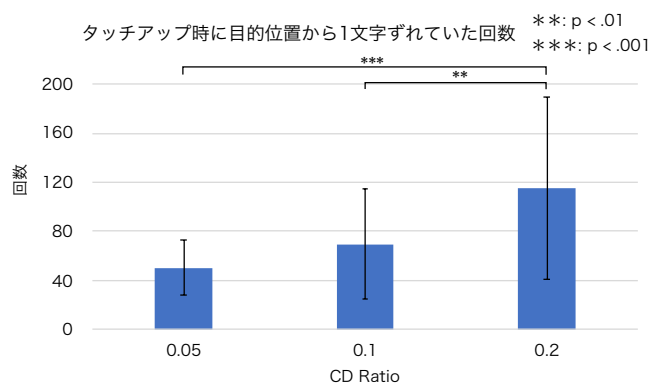


図 7.6: 100 試行あたりに指を離した際にキュレット位置が目的位置から 1 文字分ずれていた回数の平均. 誤差棒は標準偏差を表す.

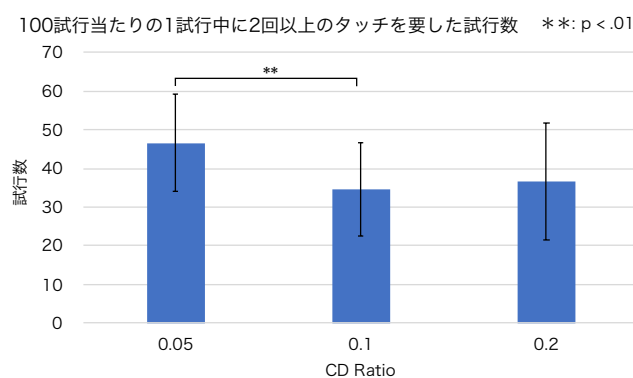


図 7.7: 100 試行中において, 2 回以上のタッチを要した試行数. 誤差棒は標準偏差を表す.

ずれるという内容のコメントがあった。さらに、スライド CD 比条件に関係なく、端のキー、特に左側のキーは指が届きにくいというコメントが複数あった。

7.3 考察

本予備実験の結果として、1セッションあたりにかかった時間、およびオーバーシュートの回数は 0.05 条件および 0.1 条件が 0.2 条件よりも良い結果であった。このため、0.05 または 0.1 がスライド CD 比に適していると考えられる。さらにこの 2 つのスライド CD 比を比較すると、図 7.4 および図 7.6 に示されるような正確性を表す指標においては 0.05 条件が 0.1 条件よりも良い。一方、図 7.5 および図 7.7 に示されるようなタッチの回数を示す指標においては 0.1 条件が 0.05 条件よりも良い。少ないタッチの回数にて処理を完了できた場合の方がユーザにとって利便性が高いと感じられるためこの指標も正確性と並び重要である。

0.05 条件および 0.1 条件においてパフォーマンスには明確な差が示されなかったため、実験協力者から得たコメントを参考にすることにした。まず、7.2.2 節においても示した通り、実験協力者から 0.05 条件においてはキャレットを多くの文字数移動させる際にイライラするとのコメントを得た。これは、実験に使用した Xperia XZ の画面ピクセル数、この端末におけるデフォルトサイズの表示文字数、およびスライド CD 比の関係に依る物だと考えられる。図 7.8 に実験に使用した端末のピクセル数を示す。この端末の横方向のピクセル数は 1080 ピクセルである。このため、指を端から端までスライドした際の最長のキャレット移動可能文字数は、0.05 条件、0.1 条件それぞれの場合 54 文字、108 文字である。また、今回実験に使用した端末における横方向の文字表示数はアルファベットにおいて最大 81 文字であった。これらの事から、スライド CD 比が 0.05 の場合、タッチスクリーンを端から端までスライドしたとしても、キャレットを 1 行分動かす事ができない。また、指を動かす際、端のキーは押しにくかったというコメントから、端から端まで自由に指をスライドする事は難しいと考えられるため、実際にはより小さな範囲しか動かす事ができないと考えられる。これらの事から、0.05 条件においては、キャレットの移動距離が長い場合に複数回のタッチを必要とするため、実験協力者からイライラするというコメントが生じたと思われる。0.05 条件においてタッチの回数が増加することは、図 7.5 から読み取ることができる。

これらの事から、我々の手法においては、ユーザの利便性を確保するため、スライド CD 比として 0.1 を採用する。しかし、本予備実験の結果はスライド CD 比として 0.1 が最適と断定できるものではないため、0.05 条件と 0.1 条件間のスライド CD 比条件をより細かく設定し、今後より詳細な分析を行う必要がある。

また、今回の実験協力者のコメントから、傾け手法を含めたソフトウェアキーボードを使った手法において、端の方のキーは使用しない事が望ましいと考えられる。さらに傾け手法の実験においてはキャレットの移動に m キーを用いていたが、キーを限定すると指の長さによって押しにくい可能性があるため、1 つのキーではなくより多くのキーを利用できるようにした方がよい可能性がある。例えば、キャレット操作をソフトウェアキーボードの 4 段目のキー全てに割り当てる事によって 4 段目のキーであれば任意のキーを押下する事によってキャレッ

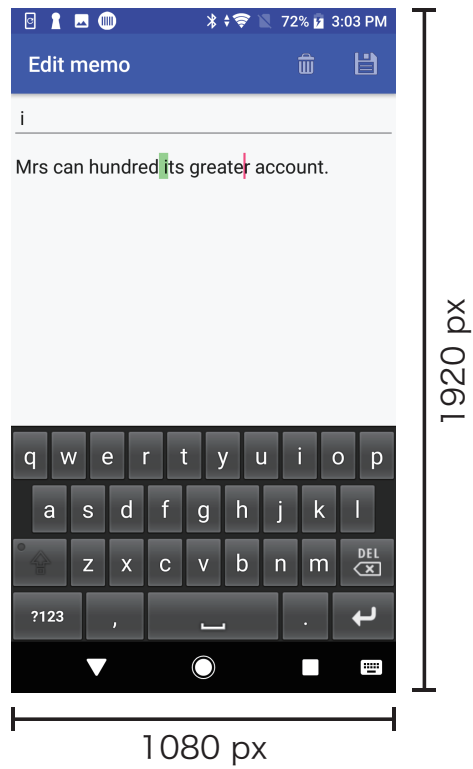


図 7.8: 実験に使用したスマートフォン (Xperia XZ) のピクセル数.

ト移動を行う事ができるようになる. 我々の提案するソフトウェアキーボードを利用した手法を設計する上ではこれらを考慮する必要がある.

第8章 手法比較実験

既存手法，傾け手法，およびスライド手法の性能を評価するため，これらを比較する実験を行った．本実験においては，5章と同じく，キャレット操作，文字列選択，および文字列操作を含めた一連の操作にかかる時間を計測し，比較した．本章にて，この実験の実験設計，実験結果，および考察を述べる

8.1 実験設計

本実験の実験協力者，実験機器，および実験タスクを述べる．

8.1.1 実験協力者

実験協力者として6名（男性5名，女性1名，平均年齢24.2歳）を雇用した．全ての実験協力者は，実験前に行ったアンケートにて，右利きまたは普段右手にてスマートフォンを使用すると回答していた．このため，今回の実験においては全ての実験協力者に右手にて実験タスクを行ってもらった．

8.1.2 実験機器

実験には本論文に示したこれまでの実験と同じく，スマートフォンとしてXperia XZを用いた．

8.1.3 実験タスク

実験協力者には，3種類の手法を用いて，キャレット操作および文字列操作に相当する操作を行ってもらった．実験協力者には着座姿勢にてこの実験を行ってもらった．図8.1に実験に使用したアプリケーションを示す．本実験にて使用したアプリケーションは5章にて使用したアプリケーションを改良したものである．5章における実験と同じく，実験アプリケーションにおいて上段に表示される文字列Aが選択対象文字列である．文字列Aは文字列Bの部分文字列となっており，実験協力者は文字列Bの中の文字列Aの部分に対して指示された手法を用いて文字列操作を行う．5章にて使用した実験アプリケーションからの変更点として，文字列Bの部分における文字列Aの部分を緑色にてハイライトする事により，選択範囲をより

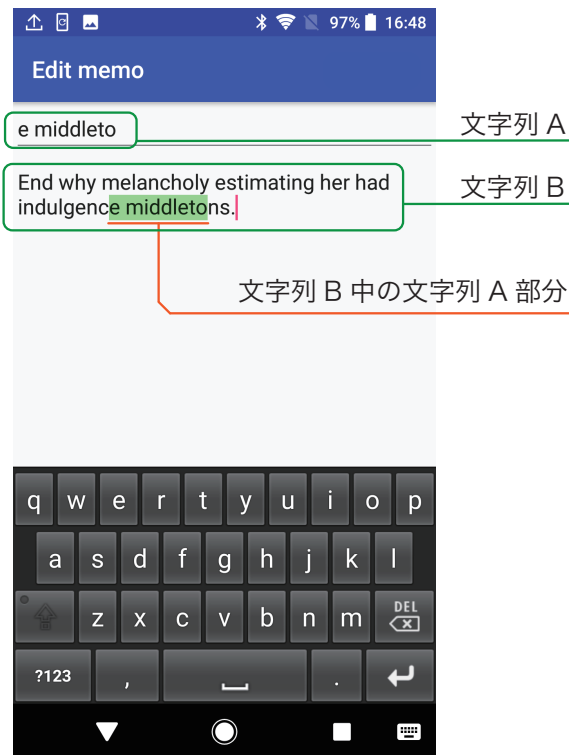


図 8.1: 手法比較実験にて使用したアプリケーション。

確認しやすくした。さらに、7章における実験時において、端に配置されているキーは使用しにくいとのコメントを実験協力者から得た。このため、使用頻度の高い機能は1つのキーではなく複数のキーを割り当てる、または指が届きやすい位置に存在するキーに割り当てべきであると考えた。このことから、本実験においては、使用頻度の高い「キャレット操作」の機能を、指が届きやすい4段目（最下段）のキーすべて（すなわち、?123 キー、';' キー、スペースキー、':' キー、およびエンタキーである）に割り当てた。このため、5章における実験とは異なり、ユーザは *m* キーではなく、4段目のいずれかのキーを押すことによってキャレット操作を行う。

以下に、本実験にて実施した3種類の文字列操作手法における操作内容を示す。

既存手法 実験協力者はまず、文字列を直接ロングタップすることによって選択状態にする。この状態にて、選択範囲を示すアジャスタをドラッグし、選択範囲を変更する。次に、文字列を選択した状態にて指を離すことにより、選択した文字列の上部に文字列操作を選択されるメニューが表示される。今回の実験においては、メニューに *Next* が表示されている。 *Next* をタップする事によって、選択されている文字列に対して文字列操作が行われる。

傾け手法 実験協力者はまず、文字列選択の開始点を変更するために、選択対象文字列の端に

キャレットを移動させる。これは、ソフトウェアキーボードの4段目のキーを押下しながらスマートフォンを傾ける事によってキャレット操作を行う事によってできる。次に、文字列の選択を行う。これは、ソフトウェアキーボードの4段目以外のキーを押下しながらスマートフォンを傾ける事によって行う事ができる。選択範囲は傾けた方向に拡張される。最後に、指を離すことによって選択されている文字列に対して文字列操作が行われる。

スライド手法 実験協力者はまず、文字列選択の開始点を変更するために、選択対象文字列の端にキャレットを移動させる。これは、ソフトウェアキーボードの4段目のキーを押下した後、指を離さずに指を画面上にてスライドする事によって行う。指をスライドした方向にキャレットが移動する。次に、文字列の選択を行う。これは、ソフトウェアキーボードの4段目以外のキーを押下した後、指を離さずに指を画面上にてスライドする事によって行う。指をスライドした方向の文字列が選択される。最後に、指を離すことによって選択されている文字列に対して文字列操作が行われる。

文字列操作を行った際に、選択対象文字列に対して正しく文字列操作が行われた場合には、文字列が更新され、次の文字列が表示される。選択されている文字列が選択対象文字列と異なっている場合には、文字列は更新されない。選択対象文字列に対して正しく文字列操作を行う事を1試行とし、11試行を1セッションとする。11試行の内、2-11試行目の10試行にかかった時間を計測した。実験協力者には手法ごとに10セッション行ってもらった。結果として時間計測を行った試行は、各実験協力者につき300試行(=10試行×10セッション×3手法)であった。また、手法間の順序効果を打ち消すために、実験協力者ごとに実施する手法の順番を変更した。疲労による影響を軽減するために、実験協力者にはセッション間に1分以上の休憩を取ってもらった。本実験は1人あたり約100分の時間を要した。実験に使用した文字列はランダムに生成されたものである¹。なお、既存手法においては、自然言語処理を用いた機能により単語の端に選択範囲が吸着するが、設定の変更は行わず、既定の仕様、つまり吸着が行われる設定にて使用した。実験協力者には各手法の全セッション終了後にSUSおよび自由記述のアンケートに回答してもらった。

8.2 実験結果

本実験において、取得したデータを用いて、操作速度およびユーザビリティに関して評価を行った。

8.2.1 操作速度

図8.2に全実験協力者における各手法ごとの1セッションあたりにかかった時間の平均を示す。図8.2に示す様に、全セッションを平均するとスライド手法における操作時間が最も短

¹<http://randomtextgenerator.com>

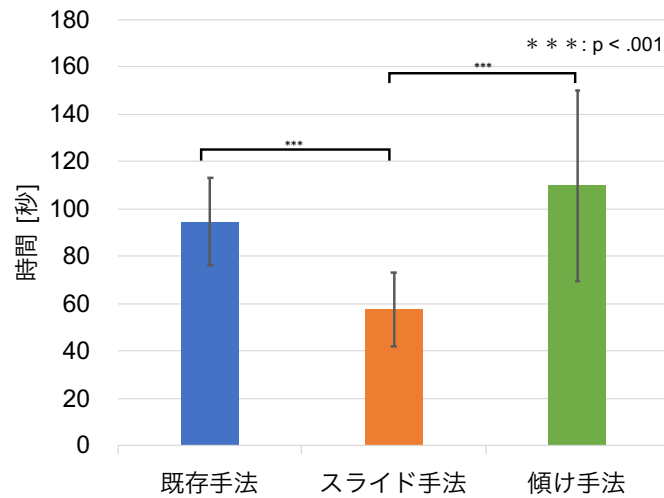


図 8.2: 全実験協力者における各手法ごとの 1 セッションあたりにかかった時間の平均. 誤差棒は標準偏差を表す.

く、傾け手法における操作時間が最も長かった. 全セッションの試行時間に関して、全ての手法において正規性がなかったため、Friedman 検定を用いて比較した. この結果、手法間に主効果が存在した ($p = 0.00 < 0.05$). さらに、Dunn らの手法 [Dun64] を用いて多重比較を行ったところ、スライド手法が既存手法および傾け手法に対して有意に試行時間が短かった (スライド手法-既存手法間: $p = 0.00 < 0.05$, スライド手法-傾け手法間: $p = 0.00 < 0.05$). また、既存手法-傾け手法間には有意差は存在しなかった ($p = 0.36 > 0.05$). また、図 8.3 にそれぞれの手法における各セッションにかかった時間の平均を示す. 図 8.3 から、傾け手法は既存手法およびスライド手法に対してセッションごとにかかった時間に関してばらつきが大きい. また、標準偏差に関しても、既存手法およびスライド手法に対して大きい. さらに、図 8.4 から、傾け手法は操作時間に関して個人差が大きい事が分かる.

8.2.2 ユーザビリティ

実験時に取得した SUS のスコアおよび実験協力者から取得したコメントを示す.

System Usability Scale

図 8.5 に手法ごとの SUS のスコアを示す. SUS のスコアはスライド手法、既存手法、傾け手法の順に高かった. 各手法ごとに取得した SUS のスコアの比較を行った. 今回取得した SUS のスコアには正規性があつたため、反復測定分散分析を行った. 結果、手法間に主効果が見られた ($p = 0.00 < 0.05$). さらに、Bonferroni 法を用いて多重比較を行ったところ、スライ

ド手法-傾け手法間にのみ有意な差が存在した ($p = 0.02 < 0.05$) . このことから, スライド手法が傾け手法に比べスコアが高いことがわかった.

コメント

実験時および実験後の自由記述アンケートにて複数の意見を実験協力者より得た. 得られた全てのコメントを表 8.1 に示す.

表 8.1: 手法比較実験にて得られたコメント.

既存手法	単語単位での選択を前提として実装されていそう (P1)
	文字単位での調整は難しかった (P1)
	端末の上部を持つ必要があり不便 (P2)
	選択が単語単位のため「it」などの単語を選びにくい (P2)
	語が指で見えなくなる (P2)
	選択範囲が指で隠れて見づらい (P4)
スライド手法	左から右に選択する方が素早く正確にできたので 最初カーソルが左側にあればありがたいかもしれない (P1)
	文字の可読性が損なわれず, 選択しやすく感じた (P2)
	割と使いやすかった (P4)
傾け手法	難しかった. 1文字ずらすような動作が欲しい (P1)
	行を超えて選択することがやや面倒 (P2)
	スマホの傾きセンサの挙動を把握するまでがむずかしい (P4)

8.3 考察

操作速度および SUS スコアにおいてスライド手法が最も良い結果であった. また, ユーザのコメントにおいても, スライド手法に対するコメントが最も好意的であったため, 今回比較した手法群においてはスライド手法が最も優れていた可能性が高い.

5章の文字列操作実験と同じく, 練習タスクを行わず本番タスクを行ったため, 全手法においてセッションあたりにかかった時間が, セッション数の進行に伴って短くなっているように考えられる. 図 8.3 下に, 20セッション目までのそれぞれの手法の近似曲線を示す. 5章における実験と同じく, 本実験においても全ての手法がセッションを重ねるごとに操作速度が上がっていくと予想される. しかし, 本実験における結果からは手法間の差は大きく変化しないと考えられる.

それぞれの手法の標準偏差を比較した所, 傾け手法の標準偏差が最も大きく, スライド手法と既存手法の差は大きくはないが, スライド手法が最も小さかった. このため, スライド手法が最も人による差異が小さく, 傾け手法が最も人による差異が大きいと感じられる. こ

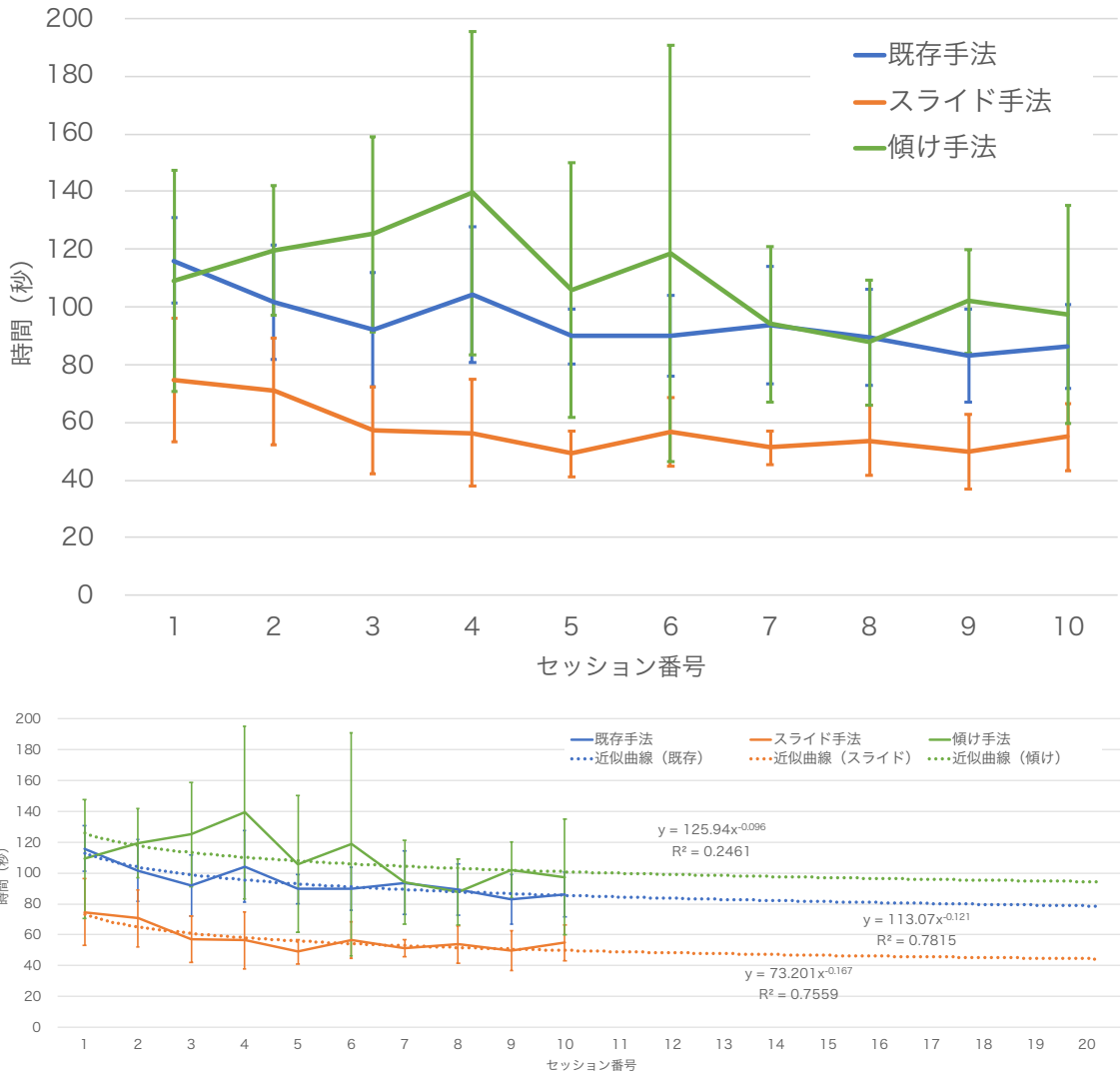


図 8.3: 操作時間の実験結果. 上: それぞれの手法における 1 セッションあたりにかかった時間の平均. 下: 上の図に近似曲線を追加し, 20 セッションまで表示したもの. 誤差棒は標準偏差を表す.

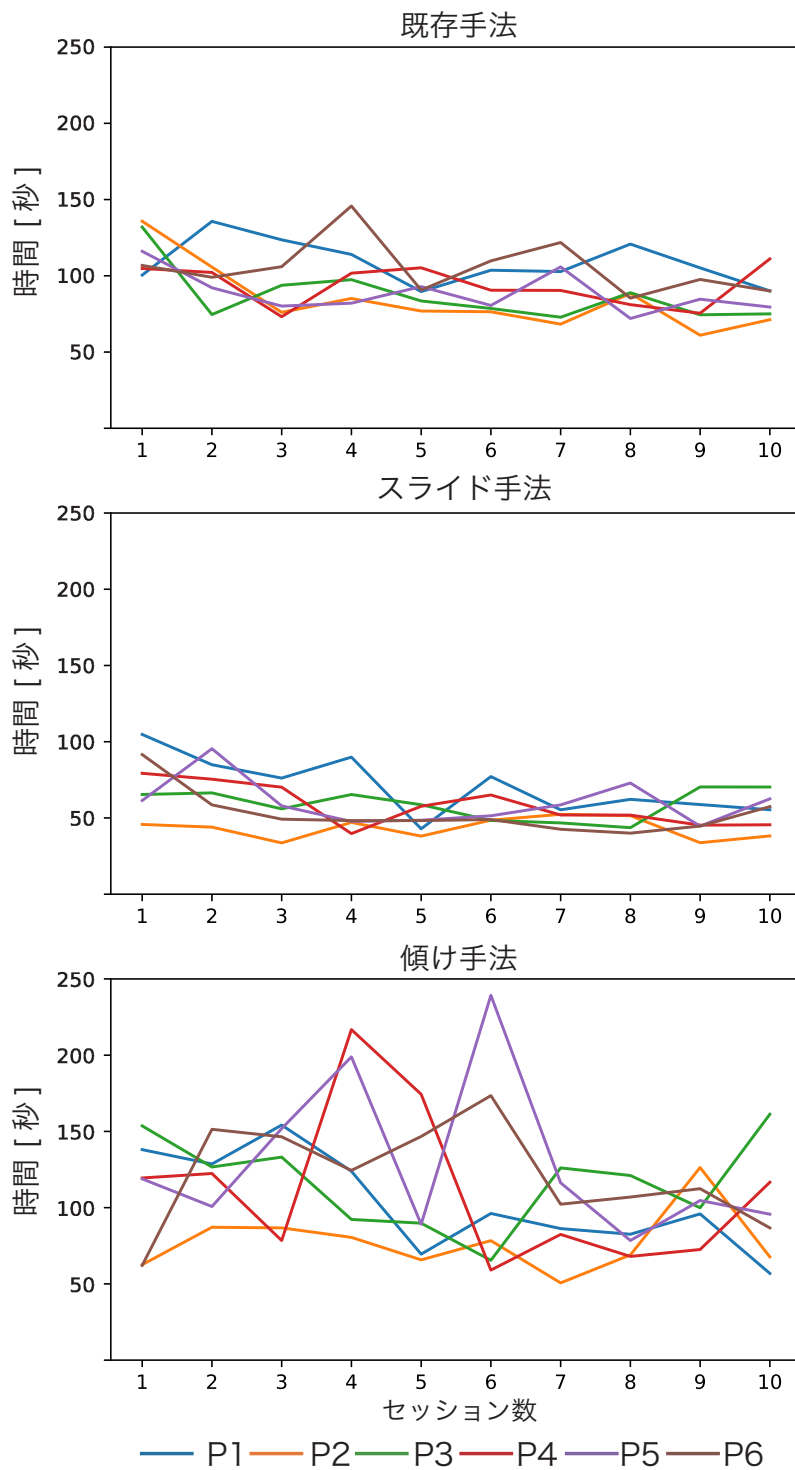


図 8.4: それぞれの手法における実験協力者ごとの各セッションにかかった時間.

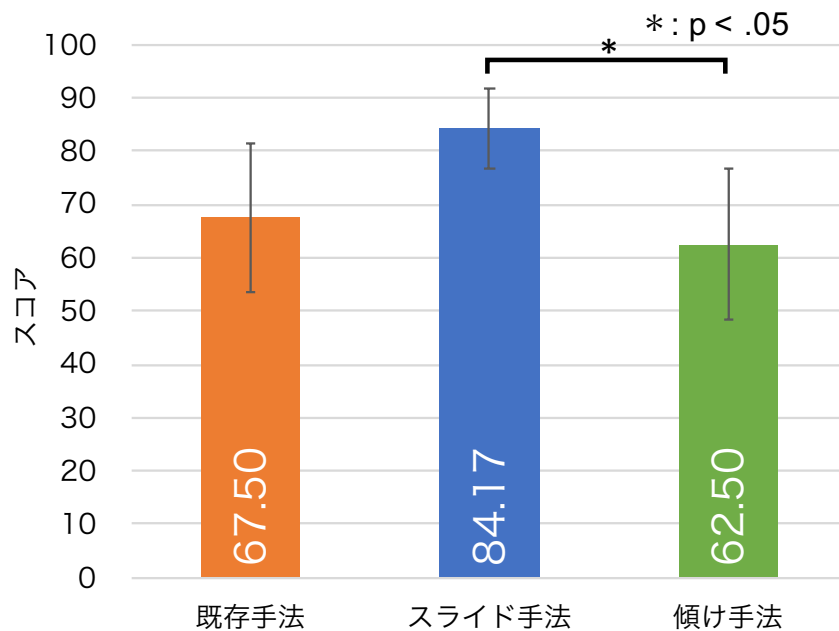


図 8.5: 手法ごとの SUS スコア. 誤差棒は標準偏差を表す.

これは、傾け手法の予備実験からも分かる様に人による傾け方の違いがあるためであると考えられる。これと比較して、画面をタッチして操作する手法は、人による感覚的な操作感の差異が小さいと考えられる。このため、スライド手法または既存手法の方が傾け手法よりもより多くの人に適している、または学習コストが低い可能性がある。

また、図 8.4 における傾け手法の結果から分かるように同じ人でもセッションによって大きくセッションにかかる時間が変化する。このことから傾け手法は一定の速度にて選択することが難しいと考えられる。しかし、同じく図 8.4 からセッションが進むにつれて、個人ごとおよび全体にてセッションにかかった時間の差が小さくなってきている事から、5 章の実験における結果と同じく傾け手法はユーザの慣れが大きく影響する手法だと考えられる。このため、今後、より長期の実験を行うことによってより正確な性能評価を行う必要があると考えられる。これに比べて、既存手法およびスライド手法は同一の実験協力者によるセッション間の差が小さい。このため、これらの手法は傾け手法と比較してより慣れやすい可能性がある。なお、この点において、既存手法は市販のスマートフォンに実装されており、実験協力者がすでにこの手法に慣れていたとも考えられる。このため、既存手法と同程度のセッション間の差が示されたスライド手法においては慣れやすさの点において他手法よりも良い可能性がある。

第9章 議論

本研究から明らかとなった今後の課題および本研究の結果を議論する。

9.1 文字による影響

本研究にて焦点を当てている文字列操作手法は、文字の大きさおよびフォントなどの文字の違いによって操作速度および操作のしやすさに影響を受けると考えられる。

9.1.1 文字の大きさ

本論文にて行った文字列操作実験では、文字の大きさを実験にて使用したスマートフォンにおける標準サイズに設定した。標準サイズ以外の文字の大きさの場合、全ての文字列操作手法において、文字列操作の行いやすさが変化するため、異なる実験結果が得られる可能性がある。先行研究 [SOSS16, SPM⁺13] においても文字の大きさが異なる条件にて実験が行われており、今後、本研究にて提案した手法を用いた場合においても異なる文字の大きさによる影響を検証する必要がある。

9.1.2 文字幅

文字の大きさと同じく、フォントによって文字幅が異なるため、これも考慮する必要がある。本研究における実験においては、Android OS 標準搭載のフォントを用いた。このフォントは文字によって横幅が異なる。このフォントに代表されるように、文字の横幅はフォントによってことなるため、同じ手法であっても、フォントが異なる場合、異なる操作感となる可能性がある。特に、等幅フォントを始めとする文字幅が均一であるフォントは、傾け手法にて実装時に考慮した、アルファベットごとの文字幅の差を考慮する必要が無い。このため、今回の実装にて採用しなかった先行研究にて使用されている手法 [SOT⁺16] を利用できる。これらの事から、文字幅を考慮した実装を行い、検証する必要がある。

9.2 ユーザの状況および周囲の環境による影響

傾け手法はユーザの状況によっては、誤動作を起こす可能性がある。例えば、ユーザが歩いているまたは走っているなど、ユーザの状況においては、スマートフォン内蔵の加速度セ

ンサが歩行または走行における加速度を検知するため、スマートフォンの傾きを正確に取得する事が難しい。また、ユーザが仰向けに寝ている状態においては、基本姿勢からの角度の変化方向が異なるため、現在の実装では提案手法が動作しないと考えられる。これと同様に、エレベータまたは電車などユーザの置かれている環境によってセンサ値が変化する場合においても予期せぬ動作が発生する。検証のために、著者がエレベータの中にて傾け手法を用いたキャレット操作を行った場合、キャレットの位置が意図に反して移動する事を確認した。このため、傾け手法においてはユーザの状況および周囲の環境からの手法への影響を調査し、環境からの影響を排除する事によってそれらの環境においても利用可能にする。

9.3 実装の改善

傾け手法およびスライド手法を用いてより素早く正確に文字列操作を行うため、実装について改良が複数考えられる。また、実験において実験協力者から複数のコメントを得たため、得られたコメントを参考として実装を改善する。

9.3.1 上下方向のキャレット操作および文字列選択

現状の実装においてはキャレット移動、および文字列選択が左右方向のみであるため、これを上下方向に関しても実装する。傾け手法の場合は、3章にて、上下方向に傾けた際の角度も取得しており、これを用いて実装する事が可能である。また、スライド手法の場合は指を上下方向にスライドする動作を上下方向のキャレット移動および文字列選択に割り当てる事によって実現できる。これらを実装することにより縦方向のキャレット移動および文字列選択が可能となれば、2行以上の文字列を対象とした操作の場合に操作速度の向上が見込める。さらに、上下方向にスクロールしながらの選択も可能となるため、より長い文章の選択が可能となる。

9.3.2 傾きを用いた文字列選択の改善

実験協力者からキャレット移動に関して「1文字ずれる事が多い」、「動きすぎる」などのキャレットの動作に関するコメントを得た。これらのコメントからキャレットの移動速度を最適化する必要がある。このため、今回の実装においては経験的に決定したキャレットの位置決定のための定数 (*cc_ratio*) に関して最適値を調査する必要がある。具体的には、傾け操作を行った回数、1回当たりの操作にかかる時間など、傾け手法に対する慣れの度合いによって動的に定数を変化させる事が考えられる。また、文字列選択に関して「空白を選択しないようにしてほしい」、「単語単位の選択がよい」など、文字列選択の効率化に関するコメントも得られた。このため、自然言語処理を行い、不必要な空白の削除や、まず単語単位の選択を行い詳細な選択を必要とする場合のみ文字単位の選択を行う段階的な選択機能を実装し、より効率的な文字列選択を実現する。

第10章 まとめ

スマートフォンを片手把持している際に、文字列選択およびコピー、カットなどのコマンド操作（文字列操作）を行うための手法を提案し、実装および評価を行った。我々の提案する手法は、ソフトウェアキーボードと、スマートフォンを傾ける動作、またはタッチスクリーン上にて指をスライドする動作を組み合わせた文字列操作手法である。我々の手法は、ソフトウェアキーボードを用いる事により、既存手法のように指を画面上部に伸ばす必要なく、スマートフォンの画面下部のみを用いて文字列操作が可能である。本論文にて、スマートフォンの傾きを利用した手法（傾け手法）および指をスライドする手法（スライド手法）を提案し、設計および実装を示した。さらに、既存手法との比較実験を実施し、操作速度およびユーザビリティの観点からこれらの手法の評価を行った。比較評価の結果、10回の文字列操作に要する時間は、既存手法、傾け手法、およびスライド手法がそれぞれ、94.6秒、109.9秒、および57.4秒であった。また、ユーザビリティの評価に利用したSUSのスコア（高い方がよい、最大値100）はそれぞれ、62.5、67.5、および84.2であった。実験の結果から、提案手法の内、傾け手法は、傾け方による個人差が大きいですが、既存手法と同等のパフォーマンスを発揮できる可能性を示した。さらに、スライド手法は、操作速度およびユーザビリティともに既存手法および傾け手法よりも高いパフォーマンスを示しており、他の手法より優れている可能性がある。

謝辞

本研究を進めるにあたり、志築文太郎先生、高橋伸先生には多くのご意見とご指導を頂きました。心から感謝致します。特に、志築文太郎先生には、常日頃より、手法の内容、実験設計、また、論文の執筆手法といった研究の事のみならず、日常生活における研究者の心構えなど多くの事をご指導頂きました。重ねて感謝致します。

インタラクティブプログラミング研究室の後輩達、同期、先輩方には様々な面において多くのご助言を頂きました。特に WAVE チームの皆様にはチームゼミにおけるご意見だけでなく、研究室にて相談に乗っていただき多くのご意見、アドバイスを頂きました。深く感謝致します。中でも、礪本俊弥氏には本研究に関する研究方針、実験設計などに関して幾度となく相談および議論をさせて頂きました。お礼申し上げます。

また、学生生活の多くの面において私を支えてくださった友人達、大学生活にてお世話になった皆様に感謝申し上げます。最後に、私を育て、様々な面において私を支えてくださった両親に深く感謝致します。

参考文献

- [AKST17] Toshiyuki Ando, Yuki Kubo, Buntarou Shizuki, and Shin Takahashi. CanalSense: Face-related movement recognition system based on sensing air pressure in ear canals. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*, UIST '17, pp. 679–689, New York, NY, USA, 2017. ACM.
- [AKST18] Toshiyuki Ando, Yuki Kubo, Buntarou Shizuki, and Shin Takahashi. CanalSense+: Face-related movement recognition and identification system based on air pressure in ear canals. In *Proceedings of the 36th Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems*, CHI EA '18, pp. VS08:1–VS08:1, New York, NY, USA, 2018. ACM.
- [AMC17] Axel Antoine, Sylvain Malacria, and Géry Casiez. ForceEdge: Controlling autoscroll on both desktop and mobile computers using the force. In *Proceedings of the 35th Annual ACM Conference on Human Factors in Computing Systems*, CHI '17, pp. 3281–3292, New York, NY, USA, 2017. ACM.
- [BLO14] Joanna Bergstrom-Lehtovirta and Antti Oulasvirta. Modeling the functional area of the thumb on mobile touchscreen surfaces. In *Proceedings of the 32nd Annual ACM Conference on Human Factors in Computing Systems*, CHI '14, pp. 1991–2000, New York, NY, USA, 2014. ACM.
- [Che15] Wei-Hung Chen. Blowatch: Blowable and hands-free interaction for smartwatches. In *Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems*, CHI EA '15, pp. 103–108, New York, NY, USA, 2015. ACM.
- [CLKS15] Youli Chang, Sehi L'Yi, Kyle Koh, and Jinwook Seo. Understanding users' touch behavior on large mobile touch-screens and assisted targeting by tilting gesture. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, CHI '15, pp. 1499–1508, New York, NY, USA, 2015. ACM.
- [CMBMS09] Andrew Crossan, Mark McGill, Stephen Brewster, and Roderick Murray-Smith. Head tilting for interaction in mobile contexts. In *Proceedings of the 11th International Conference on Human-Computer Interaction with Mobile Devices and Services*, MobileHCI '09, pp. 6:1–6:10, New York, NY, USA, 2009. ACM.

- [CMSK07] Sung-Jung Cho, Roderick Murray-Smith, and Yeun-Bae Kim. Multi-context photo browsing on mobile devices based on tilt dynamics. In *Proceedings of the 9th International Conference on Human Computer Interaction with Mobile Devices and Services*, MobileHCI '07, pp. 190–197, New York, NY, USA, 2007. ACM.
- [CPZO14] Chen Chen, Simon T. Perrault, Shengdong Zhao, and Wei Tsang Ooi. BezelCopy: An efficient cross-application copy-paste technique for touchscreen smartphones. In *Proceedings of the 2014 International Working Conference on Advanced Visual Interfaces*, AVI '14, pp. 185–192, New York, NY, USA, 2014. ACM.
- [DRPL11] Yuan Du, Haoyi Ren, Gang Pan, and Shjian Li. Tilt & Touch: Mobile phone for 3D interaction. In *Proceedings of the 13th International Conference on Ubiquitous Computing*, UbiComp '11, pp. 485–486, New York, NY, USA, 2011. ACM.
- [Dun64] Olive J. Dunn. Multiple comparisons using rank sums. *Technometrics*, Vol. 6, No. 3, pp. 241–252, 1964.
- [EG15] Alexander Keith Eady and Audrey Girouard. Caret manipulation using deformable input in mobile devices. In *Proceedings of the 9th International Conference on Tangible, Embedded, and Embodied Interaction*, TEI '15, pp. 587–591, New York, NY, USA, 2015. ACM.
- [FC09] Stephen Fitchett and Andy Cockburn. Evaluating reading and analysis tasks on mobile devices: A case study of tilt and flick scrolling. In *Proceedings of the 21st Annual Conference of the Australian Computer-Human Interaction Special Interest Group*, OzCHI '09, pp. 225–232, New York, NY, USA, 2009. ACM.
- [FIM13] Vittorio Fucella, Poika Isokoski, and Benoit Martin. Gestures and Widgets: Performance in text editing on multi-touch capable mobile devices. In *Proceedings of the 31st Annual ACM Conference on Human Factors in Computing Systems*, CHI '13, pp. 2785–2794, New York, NY, USA, 2013. ACM.
- [FM17] Vittorio Fucella and Benoît Martin. TouchTap: A gestural technique to edit text on multi-touch capable mobile devices. In *Proceedings of the 12th Biannual Conference on Italian SIGCHI Chapter*, CHIItaly '17, pp. 21:1–21:6, New York, NY, USA, 2017. ACM.
- [GCCV08] Paul Gilbertson, Paul Coulton, Fadi Chehimi, and Tamas Vajk. Using “tilt” as an interface to control “no-button” 3-D mobile games. *Comput. Entertain.*, Vol. 6, No. 3, pp. 38:1–38:13, November 2008.
- [GMG18] Alix Goguey, Sylvain Malacria, and Carl Gutwin. Improving discoverability and expert performance in force-sensitive text selection for touch devices with mode gauges.

In *Proceedings of the 36th Annual ACM Conference on Human Factors in Computing Systems*, CHI '18, pp. 477:1–477:12, New York, NY, USA, 2018. ACM.

- [GV15] Timo Götzelmann and Pere-Pau Vázquez. Inclinetype: An accelerometer-based typing approach for smartwatches. In *Proceedings of the XVI International Conference on Human Computer Interaction*, INTERACTION '15, pp. 59:1–59:4, New York, NY, USA, 2015. ACM.
- [GZMI09] Tyler J. Gunn, Hong Zhang, Ed Mak, and Pourang Irani. An evaluation of one-handed techniques for multiple-target selection. In *Proceedings of the 27th Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems*, CHI EA '09, pp. 4189–4194, New York, NY, USA, 2009. ACM.
- [HL13] Jonggi Hong and Geehyuk Lee. TouchShield: A virtual control for stable grip of a smartphone using the thumb. In *Proceedings of the 31st Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems*, CHI EA '13, pp. 1305–1310, New York, NY, USA, 2013. ACM.
- [HL15] Jaehyun Han and Geehyuk Lee. Push-Push: A drag-like operation overlapped with a page transition operation on touch interfaces. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software and Technology*, UIST '15, pp. 313–322, New York, NY, USA, 2015. ACM.
- [HPSH00] Ken Hinckley, Jeff Pierce, Mike Sinclair, and Eric Horvitz. Sensing techniques for mobile interaction. In *Proceedings of the 13th Annual ACM Symposium on User Interface Software and Technology*, UIST '00, pp. 91–100, New York, NY, USA, 2000. ACM.
- [HSM10] Guillaume Homberg, Alassane Sangott, and Benoît Martin. KeyTilt: Un clavier logiciel par inclinaison. In *Proceedings of the 22nd Conference on L'Interaction Homme-Machine*, IHM '10, pp. 137–140, New York, NY, USA, 2010. ACM.
- [Inc15] Apple Inc. 3D Touch - iOS - Apple Developer, 2015. <https://developer.apple.com/ios/3d-touch/> (2019年1月16日閲覧) .
- [Inc17] Apple Inc. Mac で Multi-Touch ジェスチャを使う, 2017. <https://support.apple.com/ja-jp/HT204895> (2018年1月16日閲覧) .
- [Inc18a] Apple Inc. iPhone User Guide - 簡易アクセス, 2018. <https://help.apple.com/iphone/11/?lang=en/iph66e10a71c#/iph66e10a71c> (2019年1月16日閲覧) .

- [Inc18b] Apple Inc. iphone でテキストを入力する・編集する, 2018. <https://support.apple.com/ja-jp/guide/iphone/iph3c50f96e/12.0/ios/12.0> (2019年1月16日閲覧) .
- [KB08] Amy K. Karlson and Benjamin B. Bederson. One-handed touchscreen input for legacy applications. In *Proceedings of the 26th Annual ACM Conference on Human Factors in Computing Systems, CHI '08*, pp. 1399–1408, New York, NY, USA, 2008. ACM.
- [KYL12] Sunjun Kim, Jihyun Yu, and Geehyuk Lee. Interaction techniques for unreachable objects on the touchscreen. In *Proceedings of the 24th Australian Computer-Human Interaction Conference, OzCHI '12*, pp. 295–298, New York, NY, USA, 2012. ACM.
- [LKB⁺18] Huy Viet Le, Thomas Kosch, Patrick Bader, Sven Mayer, and Niels Henze. Palm-Touch: Using the palm as an additional input modality on commodity smartphones. In *Proceedings of the 36th Annual ACM Conference on Human Factors in Computing Systems, CHI '18*, pp. 360:1–360:13, New York, NY, USA, 2018. ACM.
- [LMBH18] Huy Viet Le, Sven Mayer, Patrick Bader, and Niels Henze. Fingers' range and comfortable area for one-handed smartphone interaction beyond the touchscreen. In *Proceedings of the 36th Annual ACM Conference on Human Factors in Computing Systems, CHI '18*, pp. 31:1–31:12, New York, NY, USA, 2018. ACM.
- [MAQ⁺15] Sylvain Malacria, Jonathan Aceituno, Philip Quinn, Géry Casiez, Andy Cockburn, and Nicolas Roussel. Push-Edge and Slide-Edge: Scrolling by pushing against the viewport edge. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems, CHI '15*, pp. 2773–2776, New York, NY, USA, 2015. ACM.
- [NM15] Matei Negulescu and Joanna McGrenere. Grip change as an information side channel for mobile touch interaction. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems, CHI '15*, pp. 1519–1522, New York, NY, USA, 2015. ACM.
- [OO05] Ian Oakley and Sile O'Modhrain. Tilt to Scroll: Evaluating a motion based vibrotactile mobile interface. In *Proceedings of the First Joint Eurohaptics Conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems, World Haptics 2005*, pp. 40–49. IEEE, 2005.
- [PCS⁺02] Kurt Partridge, Saurav Chatterjee, Vibha Sazawal, Gaetano Borriello, and Roy Want. TiltType: Accelerometer-supported text entry for very small devices. In *Proceedings of the 15th Annual ACM Symposium on User Interface Software and Technology, UIST '02*, pp. 201–204, New York, NY, USA, 2002. ACM.

- [PKB06] Pekka Parhi, Amy K. Karlson, and Benjamin B. Bederson. Target size study for one-handed thumb use on small touchscreen devices. In *Proceedings of the 8th Conference on Human-computer Interaction with Mobile Devices and Services*, MobileHCI '06, pp. 203–210, New York, NY, USA, 2006. ACM.
- [Rek96] Jun Rekimoto. Tilting operations for small screen interfaces. In *Proceedings of the 9th Annual ACM Symposium on User Interface Software and Technology*, UIST '96, pp. 167–168, New York, NY, USA, 1996. ACM.
- [RGIS09] Mahfuz Rahman, Sean Gustafson, Pourang Irani, and Sriram Subramanian. Tilt Techniques: Investigating the dexterity of wrist-based input. In *Proceedings of the 27th Annual ACM Conference on Human Factors in Computing Systems*, CHI '09, pp. 1943–1952, New York, NY, USA, 2009. ACM.
- [RHL08] Anne Roudaut, Stéphane Huot, and Eric Lecolinet. TapTap and MagStick: Improving one-handed target acquisition on small touch-screens. In *Proceedings of the Working Conference on Advanced Visual Interfaces*, AVI '08, pp. 146–153, New York, NY, USA, 2008. ACM.
- [RT09] Volker Roth and Thea Turner. Bezel Swipe: Conflict-free scrolling and multiple selection on mobile touch screen devices. In *Proceedings of the 27th Annual ACM Conference on Human Factors in Computing Systems*, CHI '09, pp. 1523–1526, New York, NY, USA, 2009. ACM.
- [SOSS16] Kenji Suzuki, Kazumasa Okabe, Ryuuki Sakamoto, and Daisuke Sakamoto. Fix and Slide: Caret navigation with movable background. In *Proceedings of the 18th International Conference on Human-Computer Interaction with Mobile Devices and Services*, MobileHCI '16, pp. 478–482, New York, NY, USA, 2016. ACM.
- [SOT⁺16] Keigo Shima, Kazusa Onishi, Ryosuke Takada, Takuya Adachi, Buntarou Shizuki, and Jiro Tanaka. Investigating accuracy of tilting operation on wrist-worn devices with touchscreens. In *Proceedings of the 34th Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems*, CHI EA '16, pp. 2705–2711, New York, NY, USA, 2016. ACM.
- [SP09] Hamed H. Sad and Franck Poirier. Evaluation and modeling of user performance for pointing and scrolling tasks on handheld devices using tilt sensor. In *2nd International Conferences on Advances in Computer-Human Interactions*, pp. 295–300, Feb 2009.
- [SPM⁺13] Jean-Baptiste Scheibel, Cyril Pierson, Benoît Martin, Nathan Godard, Vittorio Fucella, and Poika Isokoski. Virtual stick in caret positioning on touch screens. In

Proceedings of the 25th Conference on L'Interaction Homme-Machine, IHM '13, pp. 107:107–107:114, New York, NY, USA, 2013. ACM.

- [SRC05] Katie A. Siek, Yvonne Rogers, and Kay H. Connelly. Fat Finger Worries: How older and younger users physically interact with pdas. In *Proceedings of IFIP Conference on Human-Computer Interaction, INTERACT '05*, pp. 267–280, 2005.
- [TABB14] Theophanis Tsandilas, Caroline Appert, Anastasia Bezerianos, and David Bonnet. Coordination of tilt and touch in one- and two-handed use. In *Proceedings of the 32nd Annual ACM Conference on Human Factors in Computing Systems, CHI '14*, pp. 2001–2004, New York, NY, USA, 2014. ACM.
- [TM14] Robert J. Teather and I. Scott MacKenzie. Position vs. velocity control for tilt-based interaction. In *Proceedings of Graphics Interface 2014, GI '14*, pp. 51–58. Canadian Information Processing Society, 2014.
- [VB07] Daniel Vogel and Patrick Baudisch. Shift: A technique for operating pen-based interfaces using touch. In *Proceedings of the 25th Annual ACM Conference on Human Factors in Computing Systems, CHI '07*, pp. 657–666, New York, NY, USA, 2007. ACM.
- [WB03] Daniel Wigdor and Ravin Balakrishnan. TiltText: Using tilt for text input to mobile phones. In *Proceedings of the 16th Annual ACM Symposium on User Interface Software and Technology, UIST '03*, pp. 81–90, New York, NY, USA, 2003. ACM.
- [YHHH13] Neng-Hao Yu, Da-Yuan Huang, Jia-Jyun Hsu, and Yi-Ping Hung. Rapid selection of hard-to-access targets by thumb on mobile touch-screens. In *Proceedings of the 15th International Conference on Human-computer Interaction with Mobile Devices and Services, MobileHCI '13*, pp. 400–403, New York, NY, USA, 2013. ACM.
- [YPC⁺17] Hui-Shyong Yeo, Xiao-Shen Phang, Steven J. Castellucci, Per Ola Kristensson, and Aaron Quigley. Investigating tilt-based gesture keyboard entry for single-handed text entry on large devices. In *Proceedings of the 35th Annual ACM Conference on Human Factors in Computing Systems, CHI '17*, pp. 4194–4202, New York, NY, USA, 2017. ACM.
- [大西 14] 大西主紗, 志築文太郎, 田中二郎. TouchOver:大画面を備える携帯情報端末を楽に操作するための片手親指操作手法. 第22回インタラクティブシステムとソフトウェアに関するワークショップ論文集, pp. 85–90. 日本ソフトウェア科学会, 2014.
- [島 14] 島佳吾, 箱田博之, 栗原拓郎, 志築文太郎, 田中二郎. 3点タッチを軌道ジェスチャとする範囲選択. 情報処理学会研究報告, pp. 1–8. HCI, ヒューマンコンピュータインタラクション研究会報告, 2014.

- [島 17] 島佳吾, 志築文太郎, 高橋伸. タッチパネル端末におけるオブジェクト操作のためのマルチタッチジェスチャの評価. 情報処理学会研究報告, pp. 1-8. HCI, ヒューマンコンピュータインタラクション研究会報告, 2017.

著者論文リスト

本論文に関する論文および発表

- 査読あり国際会議論文
 1. Toshiyuki Ando, Toshiya Isomoto, Buntarou Shizuki, and Shin Takahashi. Press & Tilt: One-handed Text Selection and Command Execution on Smartphone. In Proceedings of 30th Australian Conference on Human-Computer Interaction, OzCHI '18, pp.401–405, New York, NY, USA, 2018. ACM.
- 査読なし国内会議論文
 1. 安藤宗孝, 礪本俊弥, 志築文太郎, 高橋伸. スマートフォンにおける傾きを利用した文字列操作手法. 情報処理学会研究報告, Vol.2018-HCI-179, 7 pages. 2018年8月. 情報処理学会.
- その他の発表
 1. 安藤宗孝, 礪本俊弥, 志築文太郎, 高橋伸. ソフトウェアキーボードのキーに基づく文字列操作手法. 第26回インタラクティブシステムとソフトウェアに関するワークショップ (WISS2018), 2018年9月, 日本ソフトウェア科学会.

その他の論文および特許

- 査読あり国内論文誌論文

1. Ryosuke Takada, Toshiyuki Ando, Buntarou Shizuki, Shin Takahashi. BaroTouch: A Technique for Touch Force Sensing Using a Waterproof Device 's Built-in Barometer. Journal of Information Processing Vol.27 110 (Feb. 2019) (採録決定)

- 査読あり国際会議論文

1. Toshiya Isomoto, Toshiyuki Ando, Buntarou Shizuki, and Shin Takahashi. Dwell Time Reduction Technique using Fitts' Law for Gaze-Based Target Acquisition. In Proceedings of 10th ACM Symposium on Eye Tracking Research & Applications, ETRA '18, pp. 26:1–26:7, New York, NY, USA, 2018. ACM.
2. Toshiyuki Ando, Yuki Kubo, Buntarou Shizuki, Shin Takahashi. CanalSense: Face-Related Movement Recognition System based on Sensing Air Pressure in Ear Canals. In Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology, UIST '17, pp. 679–689, New York, NY, USA, 2017. ACM.
3. Toshiyuki Ando, Yuki Kubo, Buntarou Shizuki, Shin Takahashi. CanalSense+: Face-Related Movement Recognition and Identification System based on Air Pressure in Ear Canals. In Extended Abstracts of the 36th Annual ACM Conference on Human Factors in Computing Systems, CHI '18, pp. VS08:1–VS08:1, New York, NY, USA, 2018. ACM.
4. Ryosuke Takada, Wei Lin, Toshiyuki Ando, Buntarou Shizuki, and Shin Takahashi. Technique for Touch Force Sensing using a Waterproof Device's Built-in Barometer. In Extended Abstracts of the 35th Annual ACM Conference on Human Factors in Computing Systems, CHI EA '17, pp. 2140–2146, New York, NY, USA, 2017. ACM.

- 査読あり国内会議論文

1. 安藤宗孝, 久保勇貴, 志築文太郎, 高橋伸. MimiSense: 外耳道内の気圧変化を利用した下顎運動認識システム. 第24回インタラクティブシステムとソフトウェアに関するワークショップ論文集, WISS2016, pp. 1–6, 2016年12月, 日本ソフトウェア科学会.
2. 高田峻介, 林威, 安藤宗孝, 志築文太郎, 高橋伸. BaroTouch: 防水タッチパネル端末に内蔵された気圧センサを用いたタッチ圧力取得手法. 第24回インタラクティブシステムとソフトウェアに関するワークショップ論文集, WISS2016, pp. 9–14, 2016年12月, 日本ソフトウェア科学会.

- 査読なし国際会議論文

1. Toshiyuki Ando, Yuki Kubo, Buntarou Shizuki, Shin Takahashi. User Identification Method based on Air Pressure in Ear Canals. In Proceedings of Asian CHI Symposium in CHI '18, 8 pages, 2018. SIGCCHI Japan Chapter.
 2. Ryosuke Takada, Wei Lin, Toshiyuki Ando, Buntarou Shizuki, and Shin Takahashi. A Technique for Touch Force Sensing using a Waterproof Device's Built-in Barometer. In Proceedings of Asian CHI Symposium in CHI '17, 8 pages, 2017. SIGCCHI Japan Chapter.
- 査読なし国内会議論文
 1. 齋藤圭汰, 富田洋文, 安藤宗孝, 志築文太郎, 高橋伸. 抵抗測定に基づき種類識別を行うブロックシステムの検討. 情報処理学会研究報告, Vol.2019-HCI-169, 7 pages. 2016年8月. 情報処理学会.
 2. 安藤宗孝, 久保勇貴, 志築文太郎. 頭部運動に伴う外耳道内の気圧変化に関する個人差の調査. 第17回情報科学技術フォーラム講演論文集, FIT2018. pp. 3:235–3:236. 2018年9月. 電子情報通信学会.
 3. 安藤宗孝, 久保勇貴, 志築文太郎. 【既発表論文紹介】 CanalSense: Face-Related Movement Recognition System based on Sensing Air Pressure in Ear Canals. 第17回情報科学技術フォーラム講演論文集. FIT2018. pp. 3:394. 2018年9月. 電子情報通信学会.
 4. 磯本俊弥, 安藤宗孝, 志築文太郎. 近接センサおよび照度センサを用いたスマートフォンベース HMD 向けの操作手法. 第26回インタラクティブシステムとソフトウェアに関するワークショップ (WISS2018), 2018年9月. 日本ソフトウェア科学会.
 5. 岡村和典, 安藤宗孝, 志築文太郎. 入力中の文章に含まれる数値および演算子に基づいて計算結果を挿入可能なソフトウェアキーボードの開発. 情報処理学会第80回全国大会. 2 pages. 2018年3月. 情報処理学会.
 6. 久保勇貴, 安藤宗孝, 志築文太郎, 高橋伸. 能動的音響計測に基づくマイクロハンドジェスチャ認識. 情報処理学会研究報告, Vol. 2018-HCI-177, No. 32, 8 pages. 2018年3月. 情報処理学会.
 7. 高田峻介, 林威, 安藤宗孝, 志築文太郎, 高橋伸. 防水タッチパネル端末に内蔵された気圧センサを用いたタッチ圧力取得手法. 情報処理学会研究報告, Vol.2016-HCI-169, 7 pages. 2016年8月. 情報処理学会.
 - その他の発表
 1. 磯本俊弥, 安藤宗孝, 志築文太郎, 高橋伸. フィッツの法則に基づく視線を用いたターゲット選択システム. 第25回インタラクティブシステムとソフトウェアに関するワークショップ (WISS2017). 2017年12月. 日本ソフトウェア科学会.

- 出願済み特許

1. 安藤宗孝, 久保勇貴, 志築文太郎. 信号処理装置. 特願 2016-241032. 2016年12月
2. 安藤宗孝, 久保勇貴, 志築文太郎. 認証装置. 特願 2017-131440. 2017年6月

付録A 実験の際に用いた各種書類

本研究における実験にて用いた，承諾書，実験同意書，ビデオ画像の公開についての承諾書・承諾変更書を以下に示す．さらに，傾け手法の評価実験と手法比較実験において，それぞれの実験の説明のために用いた書類および事前アンケート，ユーザビリティの評価のために利用した **System Usability Scale** の記入のための用紙も合わせて示す．

A.1 承諾書

承 諾 書

平成 年 月 日

研究代表者
志築 文太郎 殿

氏 名
name

印
signature

平成30年 8月 2日付けで依頼のありましたことについて、承諾します。

基盤研究 (A)

18H04104

A.2 実験同意書

同 意 書

筑波大学システム情報系長 殿

私は、「課題名：タッチパネル搭載端末に対する操作手法の評価」について、研究概要、方法、研究対象者の必要性、研究対象者に対するリスクと安全性、研究に参加する上で想定される危険の回避、ビデオ録画を含めた個人情報の保護について十分な説明を受けました。

説明の際、本研究に協力することに同意しなくても何ら不利益を受けないこと、さらに、同意後も私自身の自由意思により不利益を受けず、参加の同意を撤回できることも理解しました。また、研究や実験の終了後であっても、枠下に指定の期日迄であれば、データ提供の同意を撤回できることも理解しました。私は、このことを理解した上で研究対象者になることに同意します。

平成 年 月 日

氏 名 _____ (署名)

「課題名：タッチパネル搭載端末に対する操作手法の評価」の研究について、次の内容について平成 年 月 日に説明を行い、上記のとおり同意を得ました。

実施責任者 所 属 _____

氏 名 _____ (署名又記名押印)

データ提供の同意撤回の期限は同意書署名の日から30日後までとさせていただきます。

研究や実験に協力した結果、不都合があった場合の連絡先

実施分担者 (所属: _____ 氏名: _____ TEL: _____)

(実施分担者が学生の場合、研究室の電話番号を記入してください。)

実施責任者 (所属:筑波大学 システム情報系 氏名:志築文太郎 TEL:029-853-5520)

筑波大学 システム情報系 研究倫理委員会 事務局

(システム情報エリア支援室 TEL:029-853-4989)

(裏面)

① 研究の概要について

私たちはスマートフォンやタブレット端末などのタッチパネルを搭載した端末に対する操作手法を開発し、その手法の性能や使用感の調査を行っています。操作手法は、タッチ、スワイプまたはドラッグといったジェスチャ入力を応用することによって、タッチパネル搭載端末の操作性の向上を狙って提案されたものです。

拘束時間

実験の1日当たりの所要時間は、休憩やアンケートの記入も含めて最大120分程度となります。

謝金の対応

実験協力者には、実験へ参加した時間に対する報酬として、筑波大学の謝金規定に基づいた時給(820円)を支払います。報酬の支給方法は口座振り込みまたはアマゾンギフト券等の提供とします。実験を途中で止めた場合も、実験協力者が実験に参加した時間から時給を算出し、支払うこととします。

研究等を行う期間

研究倫理委員会承認後 ～ 2024年 3月31日

② 研究対象者の必要性、研究への参加におけるリスクと安全性、危険回避の方法について

私たちが設計・開発したシステムによってスマートフォンやタブレットへ意図した操作が可能かどうか、そして、その使い勝手はどうかを知る必要があります。そのために、この実験では、設計・開発中のシステムを実験協力者に使用していただき、その後アンケートにご回答頂きます。回答したくない項目に関しては空欄としてください。

本実験の参加は、任意であり、参加に同意しないことによる影響は一切ありません。また、実験協力者が与えたインフォームド・コンセントについて、いつでも不利益を受けることなく撤回することができます。

本実験で用いるタッチパネル端末は、ハードウェア改造を施していない市販品のため、安全上の問題はありません。また、タッチパネル端末は実験開始前に株式会社コーヨー化成製「NB 天然アルコール除菌ウェットタオル」にて拭くことにより除菌し、実験協力者に清潔な状態にて提供します。その他実験に伴う必然的な心身上の苦痛及び不快感として、長時間の実験による拘束、及びそこから生じる疲労が考えられます。但し、ご協力いただいた時間の対価として謝金を支払い、実験中は十分な休憩を取られるようにして、それぞれ最大限の配慮を致します。

研究成果公表の際、実験協力者から得られたビデオ画像以外のデータは個人を特定できないように加工して使用させていただきます。ビデオ画像に関してはビデオ画像の公開についての承諾書の内容に従います。

研究に関する問い合わせは同意書署名欄下部に記載された連絡先(志築文太郎准教授)までお願いいたします。

③ 個人情報の保護について

- 実験中に、写真やビデオの撮影を行う場合がありますが、これらは実験の様子を記録するために行います。その写真やビデオを発表において利用する場合は、実験協力者の確認を得た上で、研究目的においてのみ利用いたします。
- 実験への参加は、協力者の自由意思によるものであり、実験への参加を随時拒否・撤回することができます。
- 学会・論文などでデータを発表する際は、データおよびそれを統計的に処理したものだけを用います。実験参加者を表現するためには、記号・数字を用います。個人を特定できる情報は公表しません。ただし、実験参加者全体については、性別の実験参加者数、年齢の範囲、所属を公表することがあります。
- 実験において得たデータは全て暗号化された状態で論文発表後 10 年間保存されます。暗号化に用いられるパスワードは本研究に関わる研究者のみが知っているものを使用します。なお、さらに安全を期すため、データに個人名が入ることはありません。個人名が記入された書類は研究期間の終了後シュレッダーにて粉碎されます。
- 上記以外のいかなる場合においても実験協力者のプライバシーは保全されます。

A.3 ビデオ画像の公開についての承諾書・承諾変更書

ビデオ画像の公開についての

承諾書・承諾変更書 (どちらかに○を付けてください。)

筑波大学システム情報系

(実施責任者) 志築文太郎 殿

私は、「研究課題：タッチパネル搭載端末に対する操作手法の評価」におけるビデオ画像が公開されることについて

(該当する選択肢の□にチェックを付けてください)

承諾します。(どちらかにチェックを付けてください。)

顔が特定される形式のまま、画像が論文発表、Web への掲載、テレビ放映などで公開されることに異存ありません。

モザイク処理等をして個人が特定されない形式にした上で、画像が論文発表、Web への掲載、テレビ放映などで公開されることに異存ありません。

以下の条件付きで公開を承諾します。

条件：(具体的に記述してください。)

承諾しません。

平成 年 月 日

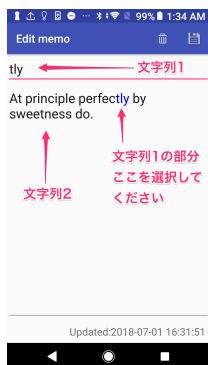
氏名 _____

A.4 実験に関する説明（傾け手法の性能評価実験）

実験に関する説明

文責：安藤宗孝

- 本実験について説明します。
 - 本実験はスマートフォンにおける文字列操作（文字列を選択してからコピー・ペーストなどの処理をする操作）に関するものです。
 - スマートフォン標準の手法と我々が開発した手法の両方を行って頂きます。
- 実験タスクについて説明します。
 1. 椅子に深く腰掛けてください。
 2. スマートフォンをお渡しするので、普段持つてるように持ってください。
 3. 下の図に示す様に、文字列1と文字列2が表示されます。
 4. 文字列2の中から文字列1の部分を指示された手法で選択してください。なお、文字列2の中の文字列1の部分は文字色を青色に変更しています。
 5. 選択後、指示された手法で操作の選択を行ってください。
 6. 選択後、次の文章が表示されます。
 - ◇ 4から6を1試行とし、10試行繰り返して頂きます。
 7. 終了後、アンケートに答えて頂きます。
 - ◇ 4から7を1セッションとし、手法毎に10セッション行って頂きます。
 8. 1セッション毎に1分程度の休憩と取るものとします。



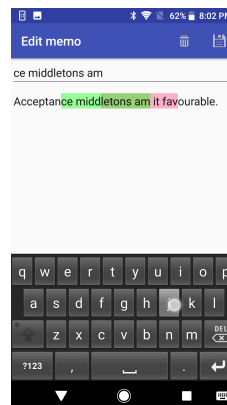
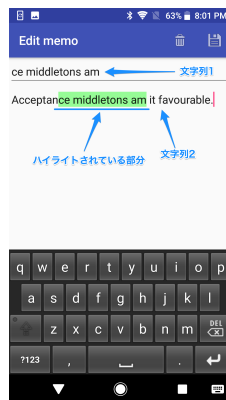
- 我々の手法について
 - カーソルの移動
 - ◇ ソフトウェアキーボードの[m]キーを押したまま左右に傾けるとカーソルが移動します。離すとカーソルがそこで止まります。
 - 文字列の選択
 - ◇ ソフトウェアキーボードの[m]キー以外のいずれかのキーを押したまま左右に傾けるとカーソルが移動し、移動した範囲を選択します。離すと現在選択されている部分の文字列に対して操作が行われます。
- 実験の際の操作
 - 従来手法
 - ◇ 青い文字の部分を実押しして選択したあと、上に出てくる[Next]を選択してください。
 - 提案手法
 - ◇ [m]キーを押して傾け、カーソルを移動してください。その後、[m]以外のキーを押して傾け、青い文字の部分を選択した後、指を離してください。
- 実験中の注意
 - できるだけ素早く正確に行ってください。
 - 実験アプリケーションが終了した、間違ってしまったなどの場合は教えてください。
 - 実験は普段スマートフォンを持つ持ち方で行ってください。

A.5 実験に関する説明（手法比較実験）

実験に関する説明

文責：安藤宗孝

- 本実験について説明します。
 - 本実験はスマートフォンにおける文字列操作（文字列を選択してからコピー・ペーストなどの処理をする操作）に関するものです。
 - スマートフォン標準の手法と我々が開発した手法（2種類）にて、文字列を選択するタスクを行って頂きます。
- 実験タスクについて説明します。
 1. 椅子に腰掛けてください。
 2. スマートフォンをお渡しするので、普段持つてるように持ってください。
 3. 下の図に示す様に、文字列1と文字列2が表示されます。
 4. 文字列2の中から緑色でハイライトされている部分（文字列2の中の文字列1の部分がハイライトされています）を指示された手法で選択してください。
 5. 選択後、指示された手法で操作の選択を行ってください。
 6. 選択後、次の文章が表示されます。
 - ◇ 4から6を1試行とし、10試行繰り返して頂きます。
 7. 終了後、アンケートに答えて頂きます。
 - ◇ 4から7を1セッションとし、手法毎に10セッション行って頂きます。
 8. 1セッション毎に1分程度の休憩と取るものとします。



- 我々の開発した手法について説明します
 - スマートフォンの傾きを用いた手法について
 - ◇ カーソルの移動
 - ✓ ソフトウェアキーボードの一番下の段のキーを押したまま左右に傾けるとカーソルが移動します。離すとカーソルがそこで止まります。
 - ◇ 文字列の選択
 - ✓ ソフトウェアキーボードの一番下段のキーを除くいずれかのキーを押したまま左右に傾けるとカーソルが移動し、移動した範囲を選択します。
指を離すと、離れた段階にて選択されている文字列に対して操作が行われます。
 - 指のスライドを用いた手法について
 - ◇ カーソルの移動
 - ✓ ソフトウェアキーボードの一番下の段のキーを押した後、指を離さずに指を画面上でスライドするとカーソルが移動します。離すとカーソルが止まります。
 - ◇ 文字列の選択
 - ✓ ソフトウェアキーボードの一番下段のキーを除くいずれかのキーを押した後、指を離さずに画面上でスライドするとカーソルが移動し、カーソルが移動した範囲が選択されます。
指を離すと、離れた段階にて選択されている文字列に対して操作が行われます。

- 実験の際の操作について説明します
 - Android 標準の手法
 - ◇ ハイライトされた文字を長押しして選択したあと、上に出てくる [Next]を選択してください。
 - スマートフォンの傾きを用いた手法
 - ◇ 一番下の段のキーを押して傾け、カーソルを移動してください。その後、一番下の段以外のキーを押して傾け、ハイライトされた文字を選択した後、指を離してください。
 - 指のスライドを用いた手法
 - ◇ 一番下の段のキーを押して、指をスライドしてカーソルを移動してください。その後、一番下の段以外のキーを押して傾け、ハイライトされた文字を選択した後、指を離してください。
- 実験中の注意
 - できるだけ 1. 素早く 2. 正確に行ってください。
 - 実験アプリケーションが終了した、間違っって行ったなどの場合は教えてください。
 - 実験は普段スマートフォンを持つ持ち方で行ってください。

A.7 実験前アンケート（手法比較実験）

● 実験前 アンケート

文責：安藤 宗孝

1. 年齢，性別，利き手についてお答えください。

年齢： ___ 歳 性別： 男・女・ほか 利き手： 左・右・両

2. 普段スマートフォンを使用する手を教えてください。

左・右・両手・時と場合による・その他 _____

3. 普段使っているスマートフォンの機種を教えてください。

_____ (例: iPhone 6S, Xperia Z5)

4. スマートフォンの利用歴を教えてください。

_____ 年 月

5. 手の長さを測定します。

手長 _____ cm 親指までの長さ _____ cm

6. 親指の幅を測定します

_____ cm

7. 実験開始時刻および終了時刻

_____ 時 分 ～ 時 分

A.8 System Usability Scale

System Usability Scale

文責：安藤 宗孝

1. このシステムをしばしば使いたいと思う
(そう思う) 5・4・3・2・1 (まったく思わない)
2. このシステムは不必要なほど複雑であると感じた
(そう思う) 5・4・3・2・1 (まったく思わない)
3. このシステムは容易に使えると思った
(そう思う) 5・4・3・2・1 (まったく思わない)
4. このシステムを使うのに技術専門家のサポートが必要とするかもしれない
(そう思う) 5・4・3・2・1 (まったく思わない)
5. このシステムにあるさまざまな機能がよくまとまっていると感じた
(そう思う) 5・4・3・2・1 (まったく思わない)
6. このシステムでは、一貫性のないところが多くあったとおもった
(そう思う) 5・4・3・2・1 (まったく思わない)
7. たいいていのユーザは、このシステムの仕様方法について、素早く学べるだろう
(そう思う) 5・4・3・2・1 (まったく思わない)
8. このシステムはとても扱いにくいと思った
(そう思う) 5・4・3・2・1 (まったく思わない)
9. このシステムを使うのに自信があると感じた
(そう思う) 5・4・3・2・1 (まったく思わない)
10. このシステムを使い始める前に多くのことを学ぶ必要があった
(そう思う) 5・4・3・2・1 (まったく思わない)

自由記述欄 (思ったことがあれば記述してください)