シングルタッチジェスチャに対する大型スマートフォン向けの片手操作手法の性能調査

八箇 恭平[1,a)]　儀本 俊弥[1,b)]　志築 文太郎[2,c)]

概要：大型のスマートフォンに対して，把持姿勢を変えることなく親指の届かない領域に対する操作を可能にするために，これまでに多くの片手操作手法が提案されている．しかし，提案されている片手操作手法は，いずれもターゲット選択（タップ）を目的に設計されている．一方で，スマートフォンの操作には，タップ以外にもスワイプ，ドラッグ，およびダブルタップ等のシングルタッチジェスチャが利用されている．そこで，我々は，全てのシングルタッチジェスチャの実行を可能にするために，カーソルを用いた 2 つの片手操作手法を設計，開発した．1 つは押下圧を利用してタッチイベントを発生させる事が可能なカーソルを用いた手法であり，もう 1 つは，タッチイベント発生位置の決定とジェスチャの実行を二段階の操作に分けたカーソルを用いた手法である．さらに，我々はシングルタッチジェスチャの実行が可能な既存手法と，提案手法の性能を調査するために実験を行った．その結果から，アプリケーションにて利用されるシングルタッチジェスチャに応じて片手操作手法を選ぶための指標が明らかになった．

1. はじめに

多くのユーザは，片手のみを用いてスマートフォンの操作を行う傾向がある [1,2]．この理由として，親指を用いたポインティングの性能が高いこと [3,4]，または片手に傘や荷物を持っている状況ではもう一方の手のみしか操作に用いることができないこと [5] が考えられる．しかし，特に iPhone XS Max（画面：6.5 inch）や Samsung S8 Plus（画面：6.2 inch）等の大型の画面を備えたスマートフォン（大型スマートフォン）においては，ユーザが把持姿勢を変えることなく親指を用いて操作できる領域は限られる [6,7]．また，親指の届かない領域に対して操作を行うためには，ユーザはスマートフォンの把持姿勢を適宜変更する必要がある．しかし，頻繁な把持姿勢の変更はスマートフォンの把持を不安定にするため，ユーザの快適な操作を妨げ [8,9]，またスマートフォンの落下を招く．

大型スマートフォンの片手操作において，把持姿勢を変えることなく親指の届かない領域に対する操作を可能にするために，多くの片手操作手法が提案されている（例：[3,10,11]）．しかし，これらの手法はいずれも親指の届かない領域に存在するターゲットを選択させること（タップさせること）を目的として設計されている．一方で，スマートフォンの操作には，タップ以外にも，スワイプ，ドラッグ，およびダブルタップ等のシングルタッチジェスチャが利用されており，片手操作時に親指の届かない領域に対してシングルタッチジェスチャの実行を可能にすること，および片手操作手法のシングルタッチジェスチャの性能調査を行うことは重要であると考えられる．

本研究において，我々は，片手操作時に親指の届かない領域に対して，把持姿勢を変更することなくシングルタッチジェスチャの実行を可能にする新しい片手操作手法として，カーソルを用いた操作手法（カーソル手法）を 2 つ開発した．先行研究においてもカーソル手法は提案されている [3, 4, 10–15]．しかし，これらの手法においては，親指が画面から離れた際にカーソルの位置にタップイベントを（すなわちタッチダウンイベントとタッチアップイベントを同時に）発生させている．そのためシングルタッチジェスチャの実行が不可能であった．一方，我々はカーソル位置にタッチイベントを発生させるための 2 種類のアプローチを考え，それぞれを用いた手法を開発した．1 つ目の手法は，押下圧を利用してタッチイベントを発生させるカーソル手法（Force Cursor）である（図 1 上）．2 つ目の手法は，タッチイベント発生位置の決定とジェスチャの実行を 2 段階の操作に分けたカーソル手法（Event Forward Cursor）である（図 1 下）．

さらに我々は，片手操作手法のシングルタッチジェス

[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
[a)] hakka@iplab.cs.tsukuba.ac.jp
[b)] isomoto@iplab.cs.tsukuba.ac.jp
[c)] shizuki@iplab.cs.tsukuba.ac.jp

© 2020 Information Processing Society of Japan

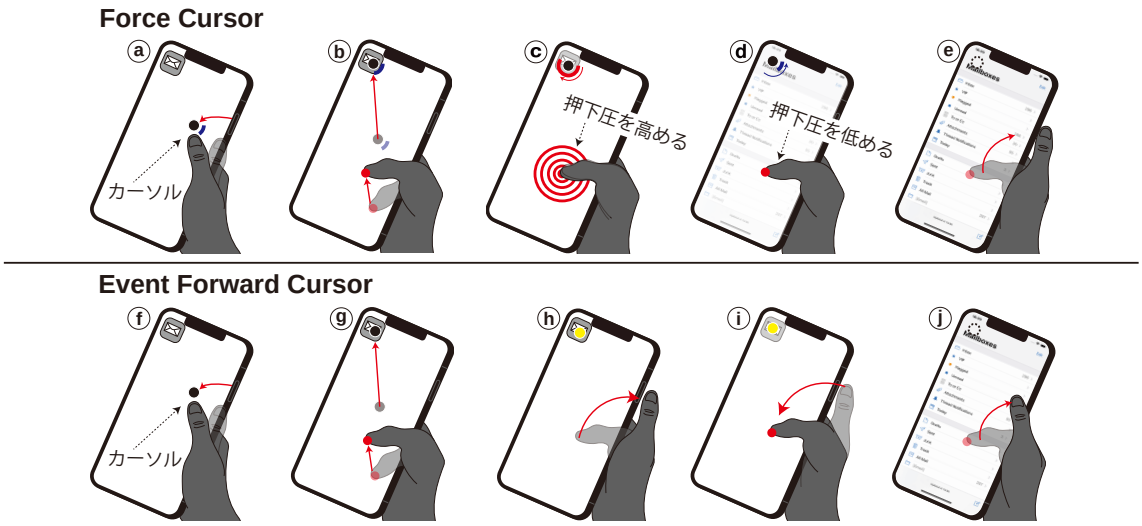

図 1: 2 つの提案手法．（上）Force Cursor．a：ベゼルからのスワイプによってカーソルモードが始まる．b：ベゼルからのスワイプに続くドラッグによってカーソルは指の移動方向に，指の移動距離 × Control-Display 比 だけ移動する．c：押下圧がある閾値以上に上がるとカーソルの位置にタッチダウンイベントが発生する．d：その後押下圧が閾値以下に下がるとカーソルの位置にタッチアップイベントが発生する．e：画面から指が離れるとカーソルモードは終了し，カーソルは画面から消える．（下）Event Forward Cursor．f，g は Force Cursor の a，b と同じである．h：カーソルモード中に指が画面から離れると，タッチイベント転送モードが始まる．i：タッチイベント転送モードにて発生したタッチイベントはカーソルの位置に転送される．j：再び指が離れるとタッチイベント転送モードは終了する．なお，タップの後，ダブルタップの待機時間（0.25 秒間）以内に再び画面に指が触れると，タッチイベント転送モードは継続する．

チャの性能を調査するために，2 つの提案手法，シングルタッチジェスチャの実行が可能な既存手法として表示を縮小する手法 [3, 16]，および親指による直接タッチという 4 手法を用いた実験を行った．この実験の結果から我々は片手操作手法の性能を議論し，さらにアプリケーションにて利用されるシングルタッチジェスチャに応じて片手操作手法を選ぶための指標を示す．

2. 関連研究

大型スマートフォンの片手操作を容易にするために，これまでに多くの片手操作手法が提案されてきた．我々は，これらの既存手法を Chang ら [3] と同じく 3 つの手法（表示を変形させる手法，代替領域を利用する手法，カーソル手法）に分類した上で本研究を位置付ける．

2.1 表示を変形させる手法

この手法は，スマートフォンの表示そのものを移動，縮小，もしくはその両方をさせることによって親指の届かない領域の操作を可能にする．

iPhone 6 以降の iPhone においては，ユーザはホームボタンをダブルタップ，もしくは画面下側から下にスワイプすることによって表示を半分下げられる [17]．また，PalmTouch [18] は，スマートフォンを把持している手の掌が画面につくと表示を半分下げる．Sliding Screen [10] は，ベゼルからのスワイプ，もしくは接触面積の大きなタッチをトリガとして起動し，親指がドラッグされると，親指の移動方向と点対称に表示を移動させる．MovingScreen [19] も同様に親指の移動方向と点対称に表示を移動させる手法であるが，ベゼルからのスワイプをトリガとして起動する際に，ユーザがベゼル上で指をドラッグした長さに応じて表示の移動速度が変化する点で異なる．TiltSlide [3] も同様に親指のドラッグによって表示を移動させる手法であり，この手法のトリガは端末を傾けることである．IndexAccess [20]，および Le ら [21] の手法においては，スマートフォンの背面にタッチパッドを取り付け，これを用いて検出した人差し指の動きによって表示を移動させる．ただし，これらの表示を移動させる手法は，表示の一部分が画面外に隠れるため，隠れた領域に対するドラッグには対応していない．一方で，我々は，全てのシングルタッチジェスチャの実行が可能な片手操作手法を提案する．

Galaxy に搭載されている One-handed Mode [16] は，ホームボタンのトリプルタップ，もしくは画面の 4 隅からのスワイプをトリガとして起動し，表示を親指の近くに縮小する．また，TiltReduction [3] も同様に，表示を縮小する手法であり，この手法のトリガはスマートフォンを傾けることである．これらの表示を縮小する手法では，全てのシングルタッチジェスチャの実行が可能であると考えられるが，これまでにその性能の調査は行われていないため，本研究にて調査する．

2.2 代替領域を利用する手法

代替領域を利用する手法とは，スマートフォンの画面上の親指の届く位置に，親指の届かない領域を操作するための領域を別に設ける手法である．

TouchOver [22] は，端末を強く振ることをトリガとして起動し，画面の下半分を上半分の代替領域として用いて，

© 2020 Information Processing Society of Japan

タッチイベントを上半分に転送する．ただし，この手法は上半分という親指の届かない領域を操作可能とするが，把持している側の反対側という親指の届かない領域には対応していない．ThumbSpace [1] は，ドラッグをトリガとして，ドラッグにて指定された大きさの領域に画面全体を縮小したポップアップを代替領域として表示する．ただしこの手法では表示全体が大きく縮小されるため，ターゲットが小さくなり，これに伴って Fat Finger 問題 [23] およびオクルージョンが発生する可能性がある．TapTap [12] も同様に画面を縮小表示したポップアップを利用するが，画面へのタッチをトリガとしており，さらにタッチされた位置の周辺のみを縮小して表示する点で異なる．またこの手法においては，ユーザは操作したい領域の周辺をタッチする必要があるため，この手法を親指の届く領域から離れた領域に用いることは難しい．提案手法は，いずれもカーソルを用いた手法であるため表示を縮小することなく片手操作を実現する．

Hasan ら [24] は，空中を代替領域として利用して，親指の 3 次元的な動きを用いた操作手法を提案している．また Lochtefeld らの手法 [25] はスマートフォン背面に取り付けたタッチパッドを代替領域として用いる．なお Yoo ら [26] は，端末の背面の人差し指を用いることによって，片手操作時の操作可能な領域を 15%拡張できることを示している．これらの操作手法 [24, 25] は，画面へのタッチを用いないため既存のジェスチャと競合が発生しない一方で，追加のデバイスが必要となる．

2.3 カーソル手法

カーソル手法とは，ユーザにターゲットを直接タッチさせる代わりにカーソルを用いて選択させる手法である．この手法の多くでは，トリガとして決められたジェスチャが行われるとカーソルを操作するためのモード（カーソルモード）が始まり，指の下にカーソルが出現する．その後，ユーザがカーソルモード中に指をドラッグすると，カーソルが指の移動距離に応じて移動する．

TiltCursor [3] は端末を傾けることを，BezelCursor [13] はベゼルからのスワイプを，Extendible Cursor [10] はベゼルからのスワイプもしくは接触面積の大きなタッチを，Extended Thumb [11] はダブルタップを，MagStick [12] はタッチをトリガとしてカーソルモードを開始する．なお，Extendible Cursor において接触面積の大きなタッチをトリガとした時にはカーソルは指の移動方向と点対称に移動し，その他の手法は指の移動方向と同方向に移動する．

CornerSpace [15] および BezelSpace [15] は指の下以外の場所にカーソルを出現させる．CornerSpace は，ベゼルからのスワイプによって画面全体を縮小したポップアップが表示される．ユーザはそのポップアップ上をタッチすることによって，画面上のタッチした位置に対応する位置にカーソルを出現させる．一方で，BezelSpace は，ベゼルからのスワイプによって表示の四隅，および中央を表すボタンが出現する．その後，ユーザはボタンを選択することによって対応した位置にカーソルを出現させる．

2D-Dragger [14]，および ForceRay [27] はカーソルの移動方法を変更した手法である．2D-Dragger は，指が移動した時に，その移動方向に存在する最も近いターゲットまでカーソルを移動する．また，ForceRay においては，ユーザが押下圧を高めるとカーソルは指から離れる方向に，押下圧を低めるとカーソルは指の方向に移動する．なお，それぞれのトリガは，画面へのタッチ，および押下圧を高めることである．これらの 2 つの手法 [14, 27] は，指の移動距離をより短くしながら親指の届かない領域への操作を可能にする．

これらのカーソル手法は，トリガ，カーソルの移動方法が異なるものの，タッチイベントを発生させるメカニズムはいずれも同じである．すなわちカーソル操作中に指が画面から離れると，カーソルの位置にタップイベントが発生する．このような，指を画面から離した時に，その時のカーソルの位置のターゲットを選択する手法は，よく利用されており，また高い選択精度となることが知られている [28]．一方で，タップ以外のシングルタッチジェスチャの実行が不可能であるという問題を伴う．我々の Force Cursor はタッチイベントを発生させるメカニズムとして押下圧を利用すること，また Event Forward Cursor はタッチイベントを発生させるメカニズムとして 2 段階の操作を利用することにより，タップ以外のシングルタッチジェスチャの実行を可能とする．

3. 提案手法の設計

我々は，片手操作時に親指の届かない領域に対して，把持姿勢を変更することなくシングルタッチジェスチャの実行を可能とする片手操作手法を 2 つ開発した．1 つは，押下圧を利用したカーソル手法（Force Cursor），もう 1 つは二段階の操作を利用してタッチイベントを転送する手法（Event Forward Cursor）である．本節においては，それぞれの設計を説明する．なお，先行研究 [3,10,27] によってベゼルからのスワイプをトリガとしたカーソル手法の性能が高いことが示されているため，我々はいずれの手法においてもカーソルモードに切り替えるためのトリガとして，ベゼルからのスワイプを採用した．また，両手法とも共通してカーソルは，親指の移動方向と同方向に移動し，カーソルの移動距離 = 親指の移動距離×Control-Display（CD）比 である．今回，我々は両手法に用いる CD 比を調べるための予備実験を行い，結果として著者らが最も使いやすいと考えた 3.0 を両手法の CD 比とした．

© 2020 Information Processing Society of Japan

3.1 Force Cursor

Force Cursor においては，ベゼルからのスワイプによってカーソルモードが始まる（図 1a）．ベゼルからのスワイプに続けてユーザがカーソルモード中に指をドラッグすると，カーソルは指と同方向に移動する（図 1b）．さらに，カーソルモード中に押下圧が，事前に決定した「タッチイベントを発生させるための押下圧の閾値」以上に高められると，タッチダウンイベントがカーソルの位置に発生し（図 1c），その後押下圧が閾値以下に低められるとタッチアップイベントがカーソルの位置に発生する（図 1d）．カーソルモードは，ユーザが指を画面から離すまで継続する（図 1e）ため，ユーザは，カーソルモード中にシングルタッチジェスチャを実行できる．例えばタップを押下圧を閾値以上に高めてから閾値以下に低めることによって，スワイプ，ドラッグを押下圧を高めた状態にて指を移動させることによって実行できる．またダブルタップも実行できる．すなわち，カーソルモード中，押下圧を閾値以上に高めてから閾値以下に低めることによってタップを実行した後，通常のダブルタップと同じく，ダブルタップの待機時間（0.25 秒間）以内に再びタップ（押下圧を閾値以上に高めてから閾値以下に低める）を実行すれば良い．今回我々は，押下圧の閾値として，ForceRay [27] にて採用されている 1.33（Nelson [29] によると 0.80 N）を用いた．

なお，押下圧を用いてタッチイベントを発生させる操作は，マウスやタッチパッドのボタンを用いた操作と近い．さらに，MacBook（Apple 社）に搭載されている Force Touch [30] においては，ユーザは Force Cursor と同様に押下圧を用いてタッチイベントを発生させられる．スマートフォンにおいても，Yong ら [31] の調査により，ユーザは押下圧を用いたクリック動作を容易に学習，実行できることが示されている．これらのことから，Force Cursor に用いられている押下圧を用いたアプローチをユーザが学習，実行することは容易であると考えられる．

また，我々の先行研究 [32,33] において，押下圧を用いたカーソル手法をユーザに体験してもらった結果から，ユーザは加えている押下圧を，フィードバック無しに把握することが困難であることがわかった．さらに押下圧を用いた操作手法においては，連続的な視覚的フィードバックが効果的であることを先行研究が示している [34–36] ため，我々も図 1a-d のように，円形バーを用いてユーザに対して連続的な視覚的フィードバックを与えた．すなわち，iPhone にて取得可能な押下圧の最大値を 100%，最小値を 0%として，押下圧が変化するたびにその押下圧の割合を円周とした円形バーが表示される．この円形バーは，押下圧が閾値以下の時には青色，閾値以上の時には赤色にて表示される．

3.2 Event Forward Cursor

Event Forward Cursor においても，Force Cursor と同様に，ベゼルからのスワイプによってカーソルモードが始まり（図 1f），このスワイプに続けてユーザが指をドラッグすると，カーソルは指と同方向に移動する（図 1g）．カーソルモード中に指が画面から離れると，モードがタッチイベント転送モードに切り替わる．タッチイベント転送モード時にはカーソルは黄色に塗りつぶされる（図 1h）．タッチイベント転送モード中には，指が画面に触れた時にタッチダウンイベントが，その後指が画面に触れている間タッチムーブイベントが，さらに指が画面から離れた時にタッチアップイベントがカーソルの位置に発生する．すなわち，ユーザは，タッチイベント転送モード中に画面上のいずれかの位置にてタップを行うとカーソルの位置にてタップを実行（図 1i），また指をドラッグするとカーソルを用いたドラッグを実行できる．タッチイベントを転送し始める位置は，カーソルモードを終了した位置となるが，この決定方法は既存のカーソル手法と同じであり，ユーザが視覚的フィードバックを受けながらイベント発生位置を決定できるため，選択精度が高いことが知られている [28]．タッチイベント転送モードは，画面から指が離れると終了する（図 1j）が，タップを転送してから，ダブルタップの待機時間である 0.25 秒以内に再び画面がタッチされると継続するため，ユーザはダブルタップやタップ&ホールド等を実行できる．なお，1 つのタッチジェスチャを転送し終えた後，再びカーソルを用いる際には，もう一度ベゼルからのスワイプを実行する必要がある．また，誤った位置にカーソルがある状態にてタッチイベント転送モードとなってしまった場合には，もう一度ベゼルからのスワイプを行うことによってタッチイベントを転送することなく，カーソルモードに再び切り替えることが可能である．

4. 実験

我々は，片手操作手法のシングルタッチジェスチャの性能を調査するために実験を行った．

4.1 参加者と使用機器

参加者は普段スマートフォンを右手にて操作している 8 人（平均 22.1 歳，全員が男性，全員が右利き）であった．実験用のスマートフォンとして，押下圧の取得が可能な大型スマートフォンである iPhone XS Max (端末：157.5 mm × 77.4 mm，画面：6.5 inch) を用いた．実験の所要時間は約 1 時間半であり，参加者は実験後に謝礼として 1660 円を受け取った．

4.2 タスク

タスクは，4 種類の手法（直接タッチおよび 3 種類の片手操作手法）を用いて 4 種類のシングルタッチジェスチャ（タップ，スワイプ，ドラッグ，およびダブルタップ）をそれぞれ片手操作にて実行することである．本実験の説明変

© 2020 Information Processing Society of Japan

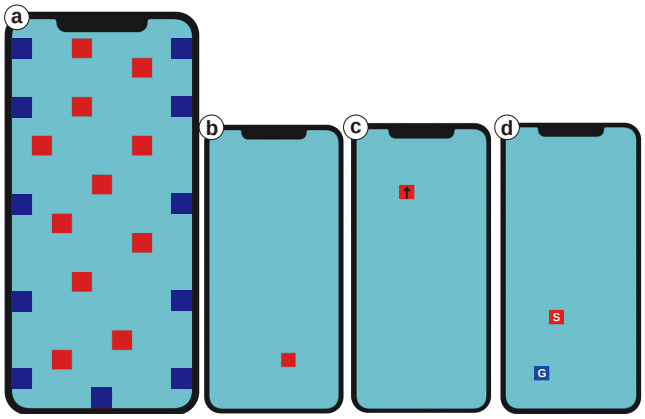

図 2: ターゲットの配置と実験中の表示．a：ターゲットの配置．ベゼルに接する 11 個のターゲット（青色のターゲット）とベゼルに接さない 11 個のターゲット（赤色のターゲット）の 2 群からなる．b：タップおよびダブルタップセッション．c：スワイプセッション．d：ドラッグセッション．

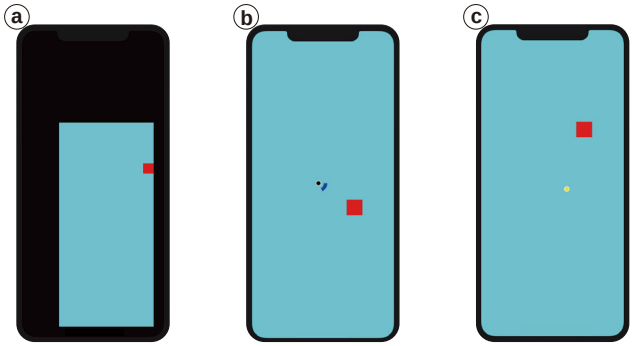

図 3: 各手法を利用した時の表示．a：RE．表示は，元々のサイズの $\frac{2}{3}$ に縮小されて，右下に寄せられる．b：FC．カーソルの周囲に押下圧を示すためのバーが表示されている．c：EC．カーソルモード時はカーソルが黒色，タッチイベント転送モードの時には黄色にカーソルが塗りつぶされる．

数は，ターゲット位置，手法，シングルタッチジェスチャの 3 つであり，それぞれの設計を以下に示す．

4.2.1 ターゲットの位置

我々はまず，iPhone XS Max の画面を $18 \times 10 = 180$ 個のセルに分割した．これは，セルのサイズが Apple の推奨最小ターゲットサイズである 6.9 mm × 6.9 mm [37] に最も近くなるように，かつ隙間なくセルを敷き詰めた時の分割である．これらの 180 個のセルのうち，ターゲットとなるセルは図 2a に示す 22 個のセルである．我々は，Corsten ら [27] の実験を参考に，ベゼルに接するターゲット（図 2a の青色のターゲット）とベゼルに接さないターゲット（図 2a の赤色のターゲット）の 2 群を設けた．ただし，Corsten らの実験においては，親指の届く領域として，右手にて片手操作を行った時の画面右下の領域に対してはターゲットが配置されなかったが，Le ら [7] は，大型スマートフォンの把持位置がユーザによって異なることを示しているため，我々は画面全体にターゲットを配置した．また，iPhone XS Max においては 4 隅のセルは一部分が欠けて表示されるため，ターゲットから除いた．なお，以降ベゼルに接するターゲットを Border，ベゼルに接しないターゲットを Center と表記する．

4.2.2 手法

実験に用いた手法は，片手操作手法を利用しない直接タッチ（DT），表示を縮小する手法（RE），および 2 つの提案手法である Force Cursor（FC）と Event Forward Cursor（EC）である．我々は，既存の片手操作手法のうち，あらゆるシングルタッチジェスチャの実行が可能な手法として RE を採用した．なお，iPhone に搭載されている手法 [17] や，Sliding Screen [10] 等の表示を移動させる手法では，一部の表示が隠れることに伴い，隠れた領域からある領域へ，もしくはある領域から隠れた領域へのドラッグが不可能となるため，実験対象には含めなかった．また我々は，片手操作手法のトリガが与える影響を除くために，3 つの手法全てのトリガをベゼルからのスワイプとした．

RE においては，ベゼルからのスワイプを行うと表示全体が縮小される．縮小スケールは，One-handed Mode [16] の標準設定である $\frac{2}{3}$ を用いた．また，縮小した表示はベゼル領域を空けて，スマートフォンの右下に配置される（図 3a）．FC および EC の設計は，前節にて説明した通りであり，それぞれ図 3b，図 3c のように表示される．

4.2.3 シングルタッチジェスチャ

シングルタッチジェスチャは，スマートフォンの操作において利用されるジェスチャとして，タップ（Tap），スワイプ（Swipe），ドラッグ（Drag），およびダブルタップ（D_Tap）の 4 種類を採用した．

タップ：参加者はターゲットに対してタップを実行する（図 2b）．なお，本実験において，タップは，タッチダウンイベントおよびタッチアップイベントが同一のターゲット上で発生した時のジェスチャとする．

スワイプ：参加者はターゲットに対して，表示されている矢印の方向にスワイプを実行する（図 2c）．スワイプ方向には，ターゲットが設定されるごとに，上下左右の 4 方向のうち 1 方向がランダムに選択された．なお，ベゼルに接するターゲットに対して，ベゼルに向かう方向のスワイプは，選択対象から外した．

ドラッグ：参加者は「S」と表示されたスタートターゲットを，「G」と表示されたゴールターゲットまでドラッグする（図 2d）．ドラッグを終了した点は，タッチアップイベントが発生した点となるため，FC または EC を用いる場合には，ゴールターゲット上にカーソルが重なっている時にタッチアップイベントを発生させると成功となる．

ダブルタップ：参加者はターゲットに対して，タブルタップを実行する（図 2a）．なお，本実験においては，ダブルタップの待機時間は iPhone の標準設定にて用いられているもの（0.25 秒）を採用した．つまり，あるターゲットに対してタップを実行してから，0.25 秒以内に同じターゲットにタップを行うと，ダブルタップとして認識される．

© 2020 Information Processing Society of Japan

4.3 手順

我々は，参加者に椅子に深く腰掛けて，右手に実験端末を把持し，右手の親指のみを用いて操作を行うように指示した．参加者は実験の説明を受けた後，最初の手法の最初のジェスチャのセッションを行った．我々は 4 つのジェスチャそれぞれに，1 つのセッションを設けた．各ジェスチャセッションにて，ランダムな順に表示される 22 個の各ターゲットに対して参加者が正しいジェスチャを実行することを 1 回の繰り返しとして，計 4 回（練習 1 回，本番 3 回）の繰り返しを行った．なお，ジェスチャの実行は全て指示した手法を用いて行われた．つまり，FC のタップセッションでは，全てのタップが FC を用いて実行された．

各繰り返しが始まると，図 2a に示した 22 個のセルのうち，タップ，スワイプ，およびダブルタップセッションにおいては 1 個のセルが，ドラッグセッションにおいては 2 個のセルがランダムに選択され，ターゲットとなる．正しいターゲットに対して，正しいジェスチャが実行されると，次のターゲットが表示される．各繰り返し中の 22 回のジェスチャ実行において，図 2a に示す 22 個全てのセルがランダムな順に一度ずつターゲットとなる．なお，ドラッグセッションにおいては，22 個のセルがランダムな順に一度ずつスタートターゲットとなり，ゴールターゲットは 22 個のセルからスタートターゲットを除いた 21 個のセルからランダムに選択される．各繰り返しが終わった後，参加者は手の疲れを取るために最低 1 分間の休憩を取った．

参加者は，各手法のタスクを全て終えた後，その手法のユーザビリティに関する調査のために，日本語に翻訳された System Usability Scale（SUS）[38] を用いてアンケートを行った．その後，次の手法にて，同様に 4 種類のジェスチャセッション，および SUS を用いてアンケートを行った．なお，我々は行われるジェスチャの順序，および使用する手法の順序をラテン方格法を用いて決定した．

4.4 結果

我々は，合計 8 参加者 × 4 手法 × 4 ジェスチャ × 22 ターゲット × 3 繰り返し = 8448 回のデータを収集した．片手操作手法の性能を比較するための目的変数として，ジェスチャ実行時間（秒），成功率（%），加速度センサ値の変化量（加加速度：$\frac{m}{s^3}$），角速度センサ値の変化量（角加速度：$\frac{rad}{s^2}$），および SUS の得点を用いた．加加速度は円滑な動きの評価に用いられる指標であり [39]，角加速度は，振動状態を評価する指標 [40] である．我々は加加速度および書く加速度を，スマートフォンの安定度を評価するために用いた．なお，スマートフォンの加速度センサ，および角速度センサは，ヨー角，ピッチ角，およびロール角それぞれの値を計測することが可能であるが，我々は 3 軸合成値を用いた．SUS の得点を除く説明変数の分析には，ターゲットの配置，ジェスチャの種類，手法を説明変数とした反復測定 3 元配置分散分析を用い，SUS の得点の分析には，手法のみを説明変数とした反復測定 1 元配置分散分析を用いた．事後検定としては Tukey の HSD 検定を用いた．いずれの検定についても有意水準は 5%である（以降の図では，$p < .05$ にて*，$p < .01$ にて**，$p < .001$ にて***と表記する）．なお，我々の目的は手法による性能の差を調査することであるため，手法に関する主効果，および交互作用についてのみを結果として述べる．

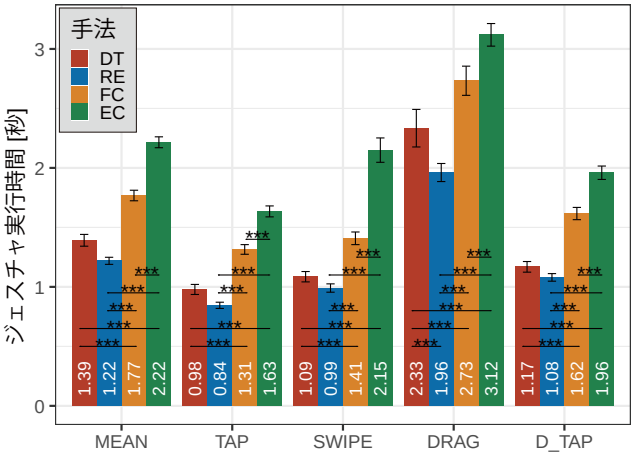

図 4: ジェスチャの実行にかかった時間（秒）．エラーバーは 95%信頼区間．

4.4.1 ジェスチャ実行時間

手法およびジェスチャ毎のジェスチャ実行時間を図 4 に示す．ジェスチャ実行時間は，あるターゲットが表示されてから，そのターゲットに対して正しいジェスチャを実行するまでにかかった時間である．手法はジェスチャの実行時間に対して有意な主効果を示し（$F_{3,8409} = 101.76, p < .001$），RE と DT 間を除く全ての間に有意な差が存在した（全て $p < .001$）．また，手法 × ジェスチャ に有意な交互作用が存在した（$F_{9,8409} = 9.71, p < .001$）．Drag セッションにおいては，全ての手法間に有意な差が存在した（全て $p < .001$）．また，他のジェスチャセッションにおいては，DT と RE を除く全ての間に有意な差が存在した（全て $p < .001$）．

DT と RE は同程度の速さであり，次いで FC，EC の順となった．EC は，他の手法と比べて Swipe セッションの実行時間が長くなった．この原因として，参加者がカーソルの操作を常にベゼル付近で行ったことが考えられる．実験参加者がモードを切り替えた後すぐにスワイプを実行しようとした結果，意図せずベゼルからのスワイプが認識されてしまう場面が何度か発生した．EC において，タッチイベント転送モード中にベゼルからのスワイプが認識されると，再びカーソルモードに切り替わる．そのため，参加者は再びタッチイベント転送モードへの移行，およびジェスチャの実行を行う必要があり，それにより時間がかかったと考えられる．

4.4.2 成功率

手法，およびジェスチャ毎の成功率を図 5 に示す．な

© 2020 Information Processing Society of Japan

お，ターゲット以外のセルにジェスチャを実行した，もしくはターゲットに対して異なるジェスチャを実行した時にエラーとなる．手法は，成功率に対して有意な主効果を示しており（$F_{3,729} = 15.75, p < .001$），DT と FC（$p < .01$），および FC と EC（$p < .001$）間に有意な差が存在した．また，手法 × ジェスチャ に有意な交互作用が存在した（$F_{9,729} = 4.75, p < .001$）．Swipe セッションにおいては DT と RE（$p < .05$），DT と FC（$p < .001$），および FC と EC（$p < .01$）間に有意な差が存在した．Drag セッションにおいては DT と RE および DT と EC 間を除く全て手法間に有意な差が存在した（RE と FC 間にて $p < .01$，RE と EC 間にて $p < .05$，その他 $p < .001$）．D_Tap セッションにおいては DT と FC（$p < .001$），RE と FC（$p < .001$），および FC と EC（$p < .001$）間に有意な差が存在した．

3 つの片手操作手法の中で最も精度が高かった手法は EC であり，ジェスチャに依らず高い成功率となった．RE は，Tap および D_Tap においては高い精度となった．しかし，Swipe および Drag においては成功率が低くなった．Tap セッションと比較して，全ての片手操作手法において Swipe セッションの成功率は下がっていることから，スワイプ実行時にはポインティング性能が低下することが考えられる．そのため，RE においてスワイプセッションの精度が低かった原因としては，ターゲットサイズがより小さくなっているため，ポインティング性能が低下する影響を強く受けたことが考えられる．また，Drag セッションは，二度のポインティング（ドラッグ元となるターゲットの選択およびドラッグ先となるターゲットの選択）が必要となるため，エラーが多くなったと考えられる．FC は，Tap，Swipe セッションの成功率は高かったものの，Drag および D_Tap においては成功率が大幅に低くなった．この原因は，押下圧を高めた状態においては指と画面間の摩擦力が大きくなるため，指を大きく動かすことが困難であったこと，押下圧の高低を素早く繰り返し切り替えることが難しかったこと，さらには普段利用しない押下圧を用いたジェスチャであったことが考えられる．

4.4.3 安定度

手法，およびジェスチャ毎の加加速度を図 6 に，角加速度を図 7 に示す．手法は，加加速度に対して有意な主効果を示し（$F_{3,8409} = 96.77, p < .001$），FC と EC を除く全ての手法間に有意差が存在した（全て $p < .001$）．また，手法 × ターゲットの位置 に有意な交互作用が存在した（$F_{3,8409} = 27.62, p < .001$）．手法 × ターゲットの位置においては，FC 時の Border と Center および EC 時の Border と Center 間を除く全ての間に主効果が存在した（全て $p < .001$）．つまり，参加者が DT および RE を用いた場合には，Border に対してジェスチャを実行した時に Center と比べて加加速度は大きかった．そのため，カーソルを用いた手法はターゲットの位置に依らず端末が安定し，一方，タッチによる選択を用いた手法はターゲットの位置によって端末の安定度が異なることがわかる．さらに，手法 × ジェスチャ にも有意な交互作用が存在した（$F_{9,8409} = 14.02, p < .001$）．手法 × ジェスチャ においては，Drag セッション時の FC と EC 間を除く全ての間に有意な差が存在した（全て $p < .001$）．

また，手法は，角加速度に対して有意な主効果を示し（$F_{3,8409} = 111.02, p < .001$），FC と EC を除く全ての手法間に有意差が存在した（全て $p < .001$）．さらに，

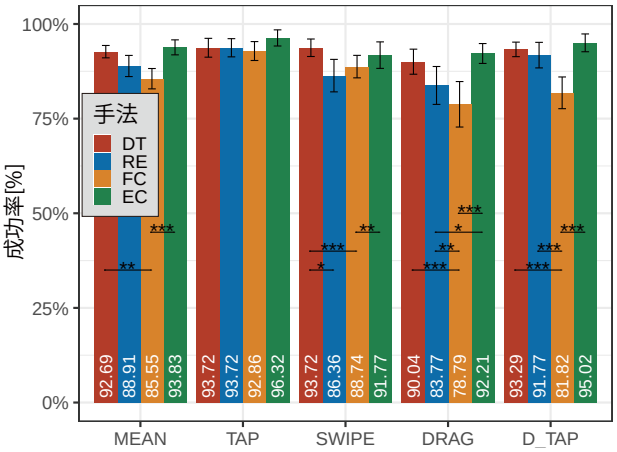

図 5: ジェスチャの成功率（%）．エラーバーは 95%信頼区間．

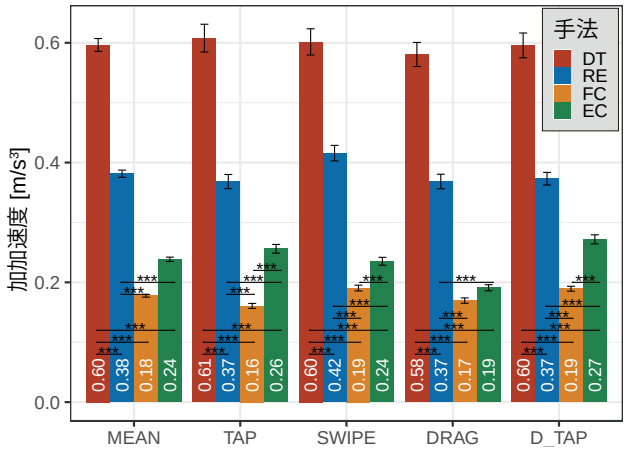

図 6: 加加速度（$\frac{m}{s^3}$）．エラーバーは 95%信頼区間．

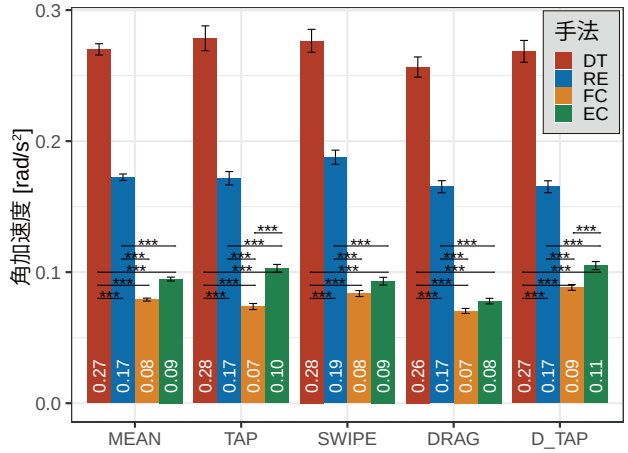

図 7: 角加速度（$\frac{rad}{s^2}$）．エラーバーは 95%信頼区間．

© 2020 Information Processing Society of Japan

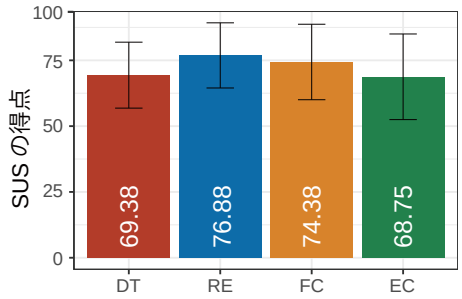

図 8: 各手法の SUS の得点．エラーバーは 95%信頼区間．

手法 × ターゲットの位置 に有意な交互作用が存在した（$F_{3,8409} = 22.23, p < .001$）．手法 × ターゲットの位置 においては，FC および EC 時の Border と Center 間を除く全ての間に有意な主効果が存在した（全て $p < .001$）．なお，手法 × ジェスチャ にも有意な交互作用が存在した（$F_{9,8409} = 13.10, p < .001$）．手法 × ジェスチャ においては，Swipe および Drag セッション時の FC と EC 間を除く全ての間に有意な差が存在した（全て $p < .001$）．

加加速度および角加速度は，FC にて最も小さく，次いで EC，RE，DT の順となった．カーソル手法においては，端末の動きが小さくなるという結果は，Corsten ら [4] がタップにて行った実験の結果と等しい．FC が EC に比べて，端末が安定した原因としては，FC を用いる場合には，ユーザが画面から親指を離す必要がないため，常に 5 本の指全てを用いて端末を把持できることが考えられる．

4.4.4 SUS の得点

SUS の得点を図 8 に示す．RE，FC，DT，EC の順に得点は高かったが，有意な主効果は存在しなかった．DT は，RE に次いで実行時間が短く，また成功率は EC に次いで高かったが，SUS の得点は低かった．この原因は，実験参加者のうち 2 人が DT の際に端末を落下させており，また 4 人の参加者が手が疲れたと述べていたことから，把持姿勢の変更が頻繁に起きていたことが考えられる．

5. 議論

本節では，実験の結果から DT および各片手操作手法の特徴について述べ，その結果から片手操作手法の選択指標について議論する．また，提案手法である FC の性能について議論する．

5.1 各片手操作手法の特徴

RE は，端末の動きをある程度許容できる場合には，有用な片手操作手法であると考えられる．この理由は，RE にてジェスチャの実行にかかる時間が DT と同程度であることである．しかし，ジェスチャ成功率に関して，タップ，ダブルタップの成功率は高かった一方，スワイプ，ドラッグの成功率は低くなった．そのため，RE は，スワイプやドラッグを利用することが少ない状況にて利用することが勧められる．このようなアプリケーションとしては，例えばスマートフォンのカメラアプリケーションが挙げられる．カメラアプリケーションにおいては，画面全体にボタンが配置されており，タップによるボタン選択を頻繁に利用する．一方で，スワイプやドラッグを利用することは少ない．

FC は，実行するジェスチャに依らず，端末を安定させた状態にて操作が可能な手法である．そのため，端末をより安定させた状態にて操作したい場合に有用な片手操作手法であると考えられる．例えば，立位時や歩行時にスマートフォンを片手操作する際には端末が不安定になると，スマートフォンの落下やそれに伴う破損の原因となるため，このような場合に FC を利用することが勧められる．しかし，今回の実験結果においては，タップ，スワイプについては成功率が高かった一方，ドラッグ，ダブルタップにおいては成功率が低かった．そのため．ドラッグやダブルタップを頻繁に用いるアプリケーションに対して FC を利用することは避ける必要がある．

EC は，実行するジェスチャに依らず，成功率が高い片手操作手法である．そのため，成功率を重視して操作を行う時や，他の手法では成功率が低かったスワイプやドラッグを利用する際に有用な手法であると考えられる．例えば，ターゲットが密集しておりポインティング精度が求められる場合，あるいはエラー修正が煩雑な場合に利用することが勧められる．一方で，EC は繰り返し利用する際には時間がかかるため，頻繁に親指の届かない領域に対する操作が発生する場合には，ユーザは煩雑に感じる可能性がある．

5.2 FC の性能の改善

我々が行った実験において，FC はドラッグ，ダブルタップ実行時の成功率が低かった．この原因は，FC を用いたドラッグにおいては，押下圧を高めた状態にて指を移動させる必要があり，摩擦力が大きくなるためユーザが高い押下圧を維持することが困難であったことが考えられる．また，この結果は，Heo らによる報告 [41,42] と同じである．そのため，ForceDrag [41] にて用いられている Force Lock のような，ドラッグ中の押下圧の変化を許容する機能を実装することによって改善される可能性がある．一方，FC を用いたダブルタップにおいては，繰り返し押下圧の高低を切り替える際に，押下圧の閾値を超えていなかったことが実験中に多く確認できた．さらに，カーソルの移動中に意図せず押下圧の閾値を超えてしまい，エラーが発生した場面も確認できたため，押下圧の微分値に対して閾値を設けた実装を行うことにより改善できる可能性がある．また，Corsten ら [4] は，押下圧を用いた手法の性能は，長期の練習によって大きく向上することを示しているため，FC の性能は，今後長期の実験にて調査する必要がある．

6. 限界と今後の課題

本節においては，今回の実験結果の考察における限界と，

© 2020 Information Processing Society of Japan

それに伴う今後の課題について述べる．

6.1 参加者の属性の偏り

今回我々が行った実験においては，参加者は 20 歳から 23 歳であり全員が若く，かつスマートフォンの利用に熟練していた．そのため，より幅広い年齢のユーザ，およびスマートフォンの利用に慣れていないユーザを募って評価を行う必要がある．特に参加者の属性が，RE および FC に与える影響は大きいと考えられる．RE においては，元々のサイズが Apple の推奨最小ターゲットサイズ（6.9 mm × 6.9 mm）となっているターゲットもさらに縮小されるため，ターゲットサイズは非常に小さくなる可能性がある．このような小さなターゲットに対しては，高いポインティング性能が求められるため年齢や経験の影響が大きくなると考えられる．また，高齢者は押下圧を用いた操作の利用が苦手であることが先行研究にて示されている [43] ことから，年齢は FC の性能に影響を与えることが考えられる．

6.2 参加者の姿勢の影響

今回我々が行った実験においては，ユーザが椅子に座っている時の片手操作について調査を行った．しかし，今後様々な状況下における各片手操作手法の性能を調査していくことが必要となる．特に，Ng ら [2] は歩行中のタップタスクを実施し，歩行中にエラー率が上昇すること，加えてもう一方の手に荷物を持っている状態においてはさらにエラー率が上昇することを明らかにしている．このような状況下においては，スマートフォンの動きが小さくなる FC や EC が有効に働く可能性がある．一方で，Wilson ら [44] が，歩行中にはユーザの押下圧の分解能が低下することを報告していることから，FC に対しても影響があることが考えられる．ただし FC に必要な押下圧の分解能は二段階であるため，歩行中においても利用できる可能性がある．

6.3 FC の押下圧の閾値の影響

今回我々が行った実験においては，FC における押下圧の閾値として，ForceRay [27] にて採用されている値を用いた．しかし，ドラッグタスクにおいては，ユーザが指を移動している途中に押下圧の高低が意図せず切り替わってしまい，エラーが発生することが何度か確認された．さらに，ダブルタップタスクにおいては，閾値を超えずに再び押下圧の高低を切り替えたためにダブルタップが実行できなかったことが頻繁に発生した．これらのことから，FC に最適な押下圧の閾値を調査する必要がある．例えば，押下圧の微分値を用いて，急激に押下圧が変化した時にタッチイベントを切り替える実装により，ユーザが指を移動している途中に意図せず押下圧の高低が切り替わってしまうエラーを取り除き，さらにダブルタップにおいても実行を容易にすることが可能であると考えられる．

6.4 端末の影響

今回我々は，実験端末として iPhone XS Max を用いた．しかし，ユーザが使用している端末のサイズや形状は様々であり，さらに端末によって各操作手法の性能が異なる可能性がある．例えば，より大きなスマートフォンを片手にて用いる際に，その端末の大きさから把持姿勢が限定されてしまい，使用できない操作手法が現れることが考えられる．そのため，今後異なる端末に対する各片手操作手法の性能を調査する必要がある．

7. おわりに

本論文においては，我々は片手操作時におけるシングルタッチジェスチャを可能とする 2 つの片手操作手法（Force Cursor，Event Forward Cursor）を提案した．また，シングルタッチジェスチャを実行する実験を行い，既存手法であった表示を縮小する手法，Force Cursor，および Event Forward Cursor の性能を調査した．その結果から，我々はスマートフォンの安定度を重視しない場合には表示を縮小する手法が，重視する場合には Force Cursor が，ジェスチャの実行精度が必要となる時には Event Forward Cursor が有効な片手操作手法となることを示した．

参考文献

[1] Karlson, A. K., Bederson, B. B. and Contreras-Vidal, J. L.: Studies in One-Handed Mobile Design: Habit, Desire and Agility, Technical report, Computer Science Dept, University of Maryland, College Park, MD (2006).
[2] Ng, A., Brewster, S. A. and Williamson, J. H.: Investigating the Effects of Encumbrance on One- and Two-Handed Interactions with Mobile Devices, CHI '14, pp. 1981–1990, ACM (2014).
[3] Chang, Y., L'Yi, S., Koh, K. and Seo, J.: Understanding Users' Touch Behavior on Large Mobile Touch-Screens and Assisted Targeting by Tilting Gesture, CHI '15, pp. 1499–1508, ACM (2015).
[4] Corsten, C., Voelker, S., Link, A. and Borchers, J.: Use the Force Picker, Luke: Space-Efficient Value Input on Force-Sensitive Mobile Touchscreens, CHI '18, pp. 661:1–661:12, ACM (2018).
[5] Karlson, A., Bederson, B. and Contreras-Vidal, J.: Understanding One-Handed Use of Mobile Devices, *Handbook of Research on User Interface Design and Evaluation for Mobile Technology*, pp. 86–101 (2008), IGL Global.
[6] Bergstrom-Lehtovirta, J. and Oulasvirta, A.: Modeling the Functional Area of the Thumb on Mobile Touchscreen Surfaces, CHI '14, pp. 1991–2000, ACM (2014).
[7] Le, H. V., Mayer, S., Bader, P. and Henze, N.: Fingers' Range and Comfortable Area for One-Handed Smartphone Interaction Beyond the Touchscreen, CHI '18, pp. 31:1–31:12, ACM (2018).
[8] Eardley, R., Roudaut, A., Gill, S. and Thompson, S. J.: Understanding Grip Shifts: How Form Factors Impact Hand Movements on Mobile Phones, CHI '17, pp. 4680–4691, ACM (2017).

© 2020 Information Processing Society of Japan

[9] Eardley, R., Roudaut, A., Gill, S. and Thompson, S. J.: Investigating How Smartphone Movement is Affected by Body Posture, CHI '18, pp. 202:1–202:8, ACM (2018).
[10] Kim, S., Yu, J. and Lee, G.: Interaction Techniques for Unreachable Objects on the Touchscreen, OzCHI '12, pp. 295–298, ACM (2012).
[11] Lai, J. and Zhang, D.: ExtendedThumb: A Target Acquisition Approach for One-Handed Interaction With Touch-Screen Mobile Phones, *IEEE THMS*, Vol. 45, No. 3, pp. 362–370 (2015), IEEE.
[12] Roudaut, A., Huot, S. and Lecolinet, E.: TapTap and MagStick: Improving One-handed Target Acquisition on Small Touch-screens, AVI '08, pp. 146–153, ACM (2008).
[13] Li, A., Fu, H. and Kening, Z.: BezelCursor: Bezel-Initiated Cursor for One-Handed Target Acquisition on Mobile Touch Screens, *Int. J. MHCI*, Vol. 8, pp. 1–22 (2016), IGI Global.
[14] Su, Q., Au, O. K.-C., Xu, P., Fu, H. and Tai, C.-L.: 2D-Dragger: Unified Touch-based Target Acquisition with Constant Effective Width, MobileHCI '16, pp. 170–179, ACM (2016).
[15] Yu, N.-H., Huang, D.-Y., Hsu, J.-J. and Hung, Y.-P.: Rapid Selection of Hard-To-Access Targets by Thumb on Mobile Touch-screens, MobileHCI '13, pp. 400–403, ACM (2013).
[16] Samsung Inc., : How Do I Use One-Handed Mode _ Samsung Support Australia (2019), `https://www.samsung.com/au/support/mobile-devices/one-hand-mode/`.
[17] Apple Inc., : Use Gestures to Navigate Your iPhone X and later - Apple Support (2020), `https://support.apple.com/en-us/HT208204`.
[18] Le, H. V., Kosch, T., Bader, P., Mayer, S. and Henze, N.: PalmTouch: Using the Palm As an Additional Input Modality on Commodity Smartphones, CHI '18, pp. 360:1–360:13, ACM (2018).
[19] Tsai, H.-R., Huang, D.-Y., Hsieh, C.-H., Huang, L.-T. and Hung, Y.-P.: MovingScreen: Selecting Hard-To-Reach Targets with Automatic Comfort Zone Calibration on Mobile Devices, MobileHCI '16, pp. 651–658, ACM (2016).
[20] Hidaka, S., Baba, T. and Haimes, P.: IndexAccess: A GUI Movement System by Back-of-Device Interaction for One-Handed Operation on a Large Screen Smartphone, *Int. J. ADADA*, Vol. 20, No. 2, pp. 41–47 (2016), Asia Digital Art and Design Association.
[21] Le, H. V., Bader, P., Kosch, T. and Henze, N.: Investigating Screen Shifting Techniques to Improve One-Handed Smartphone Usage, NordiCHI '16, pp. 27:1–27:10, ACM (2016).
[22] 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, WISS 2014, pp. 85–90, 日本ソフトウェア科学会 (2014).
[23] Siek, K. A., Rogers, Y. and Connelly, K. H.: Fat Finger Worries: How Older and Younger Users Physically Interact with PDAs, INTERACT '05, pp. 267–280, Springer-Verlag (2005).
[24] Hasan, K., Kim, J., Ahlström, D. and Irani, P.: Thumbs-Up: 3D Spatial Thumb-Reachable Space for One-Handed Thumb Interaction on Smartphones, SUI '16, pp. 103–106, ACM (2016).
[25] Löchtefeld, M., Hirtz, C. and Gehring, S.: Evaluation of Hybrid Front- and Back-of-device Interaction on Mobile Devices, MUM '13, pp. 17:1–17:4, ACM (2013).
[26] Yoo, H., Yoon, J. and Ji, H.: Index Finger Zone: Study on Touchable Area Expandability Using Thumb and Index Finger, MobileHCI '15, pp. 803–810, ACM (2015).
[27] Corsten, C., Lahaye, M., Borchers, J. and Voelker, S.: ForceRay: Extending Thumb Reach via Force Input Stabilizes Device Grip for Mobile Touch Input, CHI '19, pp. 212:1–212:12, ACM (2019).
[28] Sears, A. and Shneiderman, B.: High Precision Touchscreens: Design Strategies and Comparisons with a Mouse, *Int. J. MMS*, Vol. 34, No. 4, pp. 593–613 (1991).
[29] Nelson, R. K.: Exploring Apple's 3D Touch (2015), `https://medium.com/@rknla/exploring-apple-s-3d-touch-f5980ef45af5`.
[30] Apple Inc., : Force Touch - Apple Developer (2019), `https://developer.apple.com/macos/force-touch/`.
[31] Yong, S., Lee, E. J., Peiris, R., Chan, L. and Nam, J.: ForceClicks: Enabling Efficient Button Interaction with Single Finger Touch, TEI '17, pp. 489–493, ACM (2017).
[32] 八箇恭平, 志築文太郎, 高橋伸：大画面スマートフォンの片手操作を可能にする押下圧およびタッチ面積を用いたカーソル, WISS 2018, 2 pages, 日本ソフトウェア科学会 (2018).
[33] Hakka, K., Isomoto, T. and Shizuki, B.: One-Handed Interaction Technique for Single-Touch Gesture Input on Large Smartphones, SUI '19, pp. 21:1–21:2, ACM (2019).
[34] Cechanowicz, J., Irani, P. and Subramanian, S.: Augmenting the Mouse with Pressure Sensitive Input, CHI '07, pp. 1385–1394, ACM (2007).
[35] Wilson, G., Stewart, C. and Brewster, S. A.: Pressure-based Menu Selection for Mobile Devices, MobileHCI '10, pp. 181–190, ACM (2010).
[36] Mizobuchi, S., Terasaki, S., Keski-Jaskari, T., Nousiainen, J., Ryynanen, M. and Silfverberg, M.: Making an Impression: Force-controlled Pen Input for Handheld Devices, CHI EA '05, pp. 1661–1664, ACM (2005).
[37] Apple Inc., : Adaptivity and Layout - Visual Design - iOS - Human Interface Guidelines - Apple Developer (2018), https://developer.apple.com/design/human-interface-guidelines/ios/visual-design/adaptivity-and-layout/.
[38] Brooke, J.: SUS-A Quick and Dirty Usability Scale, *Usability Evaluation in Industry* (1996), Taylor & Francis.
[39] Lank, E., Cheng, Y.-C. N. and Ruiz, J.: Endpoint Prediction Using Motion Kinematics, CHI '07, pp. 637–646, ACM (2007).
[40] Zhao, H. and Feng, H.: A Novel Angular Acceleration Sensor Based on the Electromagnetic Induction Principle and Investigation of its Calibration Tests, *Sensors*, Vol. 13, No. 8, pp. 10370—10385 (2013), MDPI.
[41] Heo, S. and Lee, G.: ForceDrag: Using Pressure as a Touch Input Modifier, OzCHI '12, pp. 204—207, ACM (2012).
[42] Heo, S. and Lee, G.: Force Gestures: Augmenting Touch Screen Gestures with Normal and Tangential Forces, UIST '11, pp. 621–626, ACM (2011).
[43] Kinoshita, H. and Francis, P. R.: A Comparison of Prehension Force Control in Young and Elderly Individuals, *Eur. J. Appl Physiol Occup Physiol*, Vol. 74, No. 5, pp. 450–460 (1996), Springer.
[44] Wilson, G., Brewster, S. A., Halvey, M., Crossan, A. and Stewart, C.: The Effects of Walking, Feedback and Control Method on Pressure-based Interaction, MobileHCI '11, pp. 147–156, ACM (2011).

© 2020 Information Processing Society of Japan