

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

大西 主紗 志築 文太郎 田中 二郎*

概要. 大型のタッチ画面を備える携帯情報端末（以降、大端末）を片手にて操作する場合、その大きさゆえに指が届かず、操作が困難な画面領域が存在する．この時ユーザは片手のみを用いて端末の把持姿勢を変えるという煩雑な動作を行う必要があり、不安定な把持及び端末の落下の原因となる．そこで我々は、大端末向けの片手親指操作手法である TouchOver を示す．TouchOver は画面下部にて発生したタッチイベントを画面上部に転送する．画面上部に対する TouchOver と画面下部に対する直接操作とを使い分けることにより、大端末に対する楽な操作、すなわち片手親指のみを用いた持ち替えを必要としない、画面の全領域に対する操作を実現する．加えて操作の切り替えに端末をシェイクするという既存のタッチジェスチャとの競合が発生しない動作を用いる．我々は TouchOver のプロトタイプを Android アプリケーションとして実装した．本稿ではプロトタイプを用いた使用例も示す．

1 はじめに

Parhi らによれば、多くのユーザは携帯情報端末の片手操作、すなわち端末を把持した手の親指のみを用いた操作を欲している [11]．なぜならば、携帯情報端末のブラウザを用いて「ちょっと調べ物をする」際や、携帯情報端末を音楽プレーヤとして使用する場合などは、片手のみを用いた操作が非常に便利だからである．この欲求は、傘やつり革などで片手がふさがっている場合や片手をけがしていて両手による操作が困難な場合に誰にでも生じ（図 1a）、また、couch potato と呼ばれる、常にソファに寝そべっているようなものぐさなユーザにおいてはより顕著であると考えられる（図 1b）．

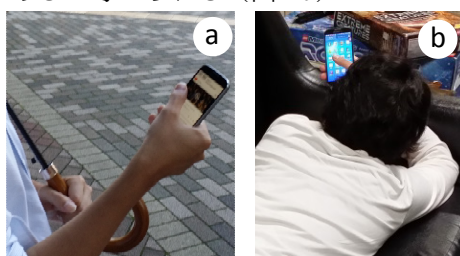


図 1. 片手による携帯情報端末操作が要求される場面。
a：片手がふさがっている場合．b：ソファに寝転がっている場合．

一方で、大型のタッチ画面をもつ携帯情報端末（以降、大端末）が市場に登場し、ユーザに使用されるようになった．大端末にはその画面の大きさから一度に表示可能な情報量が増加するというメリットが

存在する一方、片手操作によって大端末を操作する場合にユーザの指が届かない画面領域に対する操作が困難であるという問題が存在する [17]．この場合ユーザは片手のみを用いて大端末の把持姿勢を変える必要がある．しかし、その動作は煩雑であり、かつ不安定なものであるため端末の落下の原因となる．

本稿において、我々は大端末向けの片手親指操作手法である TouchOver を示す．この手法は大端末に対する楽な操作、すなわち片手親指のみを用いた持ち替えを必要としない、画面の全領域に対する操作手法である．本手法は図 2 に示すように、画面下部にて発生したタッチイベントを画面上部に転送することによって、ユーザの親指が届く大端末の画面下部のみを用いて画面の全領域に対する操作を可能にする．本手法を用いた場合、ユーザは大端末の把持姿勢を変えることなくポインティングを含む既存のタッチジェスチャ全般を画面の全領域に対して行うことが可能であるため、常に端末を安定して把持し続けることが可能である．

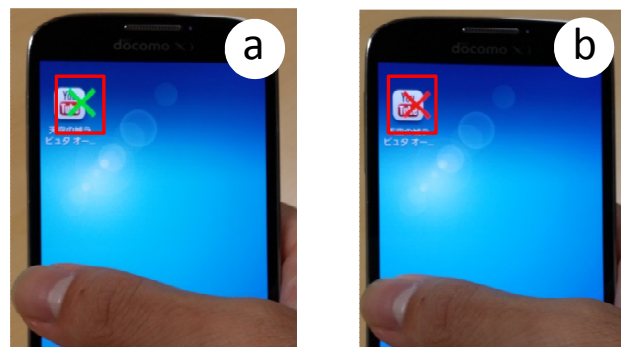


図 2. TouchOver 使用時に表示されるフィードバック．

Copyright is held by the author(s).

* Kazusa Onishi, 筑波大学大学院システム情報工学研究科 コンピュータサイエンス専攻, Buntarou Shizuki and Jiro Tanaka, 筑波大学 システム情報系

なお本手法はタッチイベントの転送予想点をホバーイベントを活用することにより分かりやすく表示する．具体的には，ホバーイベントが発生している際には転送予想点を緑色のカーソルとして表示し（図 2a），画面がタッチされている際にはカーソルの色を赤色に変化させる（図 2b）．これによりユーザはタッチ前に「このままタッチすると画面上部のどこをタッチしたことになるのか」，またタッチ後に「現在は画面上部のどこを操作していることになっているのか」を把握することが可能となるため，確実に操作を行うことが可能である．現在ホバー検出が可能なスマートフォンには ELUGA P P-03E, Galaxy S4 SC-04E, AQUOS Phone ZETA SH-06E, ARROWS NX F-06E 等が存在するが，タッチイベントの転送自体にはホバーイベントを必要としないため，フィードバックを用いることなく転送予想点を把握できる場合には習熟により本手法は使用可能である．

本手法では，ユーザが画面の全領域に対する操作を画面下部にて行うため，画面上部への間接操作と画面下部への直接操作をユーザが使い分けの必要がある．そこで本手法では，これらの操作の切り替え，すなわち TouchOver の起動/終了に端末をシェイクする動作を用いる．シェイクは素早く特定の機能を実行可能な動作として Wu らによっても用いられており [16]，頻繁に TouchOver の起動/終了を行う本手法に適する．シェイク動作は端末に組み込まれている加速度センサのみにて検出することが可能であり，また既存のタッチジェスチャとの競合を避けることが可能である．

2 関連研究

大端末および大画面デスクトップ環境などの大領域を対象とする操作手法が多く存在する．本節ではそれらの間接/直接操作手法と比較しながら，間接操作手法である TouchOver の立ち位置を述べる．また，本手法は TouchOver の起動/終了にシェイク動作を用いているため，タッチパネル端末向けの操作切り替え手法についても述べる．

2.1 大領域向けの間接/直接操作手法

間接操作手法を用いて親指の届かない領域に対する操作を実現した手法が多く存在する．MagStic [14] は親指の動きと逆方向に移動し，ターゲットに吸い付くカーソルである．CornerSpace [17] は，Bezel-Swipe [13] を用いて出現させた UI によって出現位置を画面 4 隅の中から任意に選択できるカーソルである．ただし，これらの手法では，ユーザはタッチアップが行われた際にカーソルのある位置に対してポインティングを行うのみであり，ドラッグやロングタップなどのタッチジェスチャを行うことはできない．ExtendedThumb [10] も上記の手法と同様に

ドラッグによってカーソルを親指の届かない領域に対して移動させるが，カーソルを移動させタッチアップを行った後に続くタッチ操作がすべてカーソルの位置に対して転送される．これにより親指の届かない領域に対する操作を実現しているが，1 つの操作に対してカーソルの移動および領域に対する操作という 2 段階のタッチ操作を毎回行わなければならない．本手法はポインティングのみならず，ダブルタップやロングタップなどのタッチジェスチャ全般を大端末に対して使用可能にする．また，TouchOver は起動/終了，及び大端末の操作という 2 段階の操作を必要とするが，1 度起動した TouchOver は再び終了させるまでタッチイベントを画面上部に対して転送し続ける．したがって操作対象が画面上部に集まっている場合は ExtendedThumb よりも少ない操作量にて操作可能である．また，操作に用いる指によって操作対象が隠れる問題を対象とした，カーソルを用いる携帯情報端末向け操作手法の研究も存在する．Shift [15] や Tyler らの研究 [4] においては，タッチを行った親指によって隠れた画面が親指の外側に表示され，ユーザはドラッグ操作にて移動可能なカーソルを用いて親指の下のターゲットを選択する．間接操作である本手法も，これらの手法と同様に指によって操作対象が隠れる問題を解決すると考えられる．

一方，直接操作手法を用いて大端末の片手操作を実現した手法も存在する．ThumbSpace [8] は，大端末画面全体の縮小画像が表示されたタッチパッドを生成する．ユーザはこのタッチパッドに表示されたターゲットをタッチすることにより，指が直接届かない画面端のターゲットを選択することができる．ThumbSpace は画面領域全体のタッチパッドを生成するのに対し，TouchOver は画面下部をタッチパッドとして用いることにより画面上部の操作を容易にしている．Drag-and-Pop や Drag-and-Pick [1]，Vacuum [2] は，画面上に存在するアイコンやウィンドウをユーザの近くに移動させる大画面向けの操作手法である．ただし，Drag-and-Pop 及び Drag-and-Pick ではアプリケーションアイコンに対するファイルアイコンのドロップやフォルダの展開などアイコンに対する操作しか行えず，Vacuum では手元に移動したウィンドウの中からユーザが選択したものを手元に残すことしかできないため，ユーザは手の届かない領域の UI を操作することができない．LoopTouch [18] はロールジェスチャと呼ばれる，大端末の表面に触れている親指と背面に触れている人差し指の相対位置が近づくようなジェスチャを行うことにより，画面自体をループさせ操作対象を親指の近くに移動させる．この手法は画面のレイアウトを変化させるものであるため，本来画面下部に配置されているアイコンが画面上部に移動するなど，ユーザが混乱させられる可能性がある．一方本研究は画

面のレイアウトに一切の変化を加えず、かつその操作はユーザがマウス操作において慣れ親しんだ間接操作と似たものとなっている。

2.2 タッチパネル向けの操作切り替え手法

携帯情報端末に対する特殊な動作を、端末の操作の切り替えに用いる手法が研究されている。GripSense [3] は端末に組み込まれたジャイロセンサ及び端末のバイブレーション機能を用いて端末に対する握りの強弱を判定し、携帯情報端末の操作を切り替える。同様に Force Gestures [6] も、端末に取り付けた圧力センサを用いて端末に対する握りの強弱を判定している。これらの手法を用いて TouchOver の起動/終了を行うことは可能である。ただし、本手法は大端末を楽に操作することが主眼となっているため、力を込める必要があるこれらの手法ではなくシェイク動作を操作の切り替えに用いる。

また、Forcetap [7] は携帯情報端末に組み込まれている加速度センサを用いて、Expressive Touch [12] はテーブルトップの 4 隅に取り付けられたマイクを用いてタップの強弱を判定した。TapSense [5] は携帯情報端末に張り付けたマイクを用いて、画面に触れた指の部位を判定した。これらの手法はそれぞれ、特殊なタッチダウンが発生した際に操作を切り替えるべきか否かを判定している。本手法においてもこれらの手法を用いることによって、特殊なタッチダウンが発生した際には TouchOver による画面上部の操作を、従来通りのタッチダウンが発生した際には従来通りの直接操作を切り替えて使用することは可能である。しかし、タッチダウンの際に特殊な操作を行うことは端末の安定した把持に影響し、操作精度が悪化する要因となりうる。

3 TouchOver

TouchOver は大端末向けの片手親指操作手法であり、その要件は大端末に対する楽な操作、すなわち片手親指のみを用いた持ち替えを必要としない、画面の全領域に対する操作を可能とすることである。そこで本研究においては以下の 2 点をアプローチとする。

- ユーザは指が届かない領域に対しては間接操作を用いて操作を行う（間接操作の設計）。
- 間接操作と直接操作とを任意に切り替える（切り替え動作の設計）。

3.1 間接操作の設計

本手法にて用いる間接操作は、画面下部のみを用いた画面上部の操作である。従来の間接操作手法の多くは画面に対するタッチダウンを用いてカーソルを出現させ、ドラッグ操作を用いてそのカーソルを目的の位置に移動させ、タッチアップにてカーソル

が存在する位置をポインティングさせる。しかし、これらの手法はドラッグやロングタップといったタッチジェスチャを行うことができず、また、カーソルを動かすためのドラッグ操作と端末において動作しているアプリケーションに対するドラッグ操作（例：スクロール）が競合する。

このような競合は操作の煩雑化の原因となりうるため、TouchOver においては、画面下部にて発生したタッチイベントを画面上部に転送する手法を用いる。すなわち画面の縦の長さを h としたとき、座標 (x, y) にて発生したタッチイベントを座標 $(x, y - h/2)$ に転送する。これにより、画面下部のみを用いてジェスチャを含む全ての操作を画面上部に対して行うことができる。なお、 $y < h/2$ の領域、すなわち画面上部にタッチダウンが発生した場合、TouchOver が起動している場合においてもその一連の操作（例：画面上部から画面下部へのドラッグ操作）を直接操作として扱う。

この手法を用いる場合、特に小さいターゲットを操作する際には、タッチダウン時にタッチイベントがどこに転送されるかのフィードバックが必要となる。そこで本手法においては、TouchOver 起動時にホバーイベントが発生している際には図 2a のようにタッチイベントの転送予想点を緑色のカーソルとして表示する。ホバーイベントとは、端末の画面から少し離れた位置に指があるときに発生するイベントである。図 3 に示すように、ホバーイベント発生時の指を画面に対して垂直におろした場合、ホバーイベントの発生した座標およびタッチイベントの発生する座標が一致する。すなわち、ホバーイベントが座標 (x_{hover}, y_{hover}) に発生している際には、座標 $(x_{hover}, y_{hover} - h/2)$ をタッチイベント転送予想点のフィードバックとして利用することが可能となる。またタッチイベントが発生している際には、タッチイベントの転送先に赤色のカーソルを図 2b のように表示することにより、タッチイベントが転送されている位置をフィードバックする。

このフィードバックにより、ユーザはホバーしている親指の移動を用いて先述の間接操作手法のように TouchOver を使用することができる。したがって、TouchOver は従来のカーソルを用いた間接操作手法と同程度の容易さで学習が可能である。

3.2 切り替え動作の設計

本手法においては、直接操作と間接操作を切り替える動作として、大端末をシェイクする動作を用いる。これは 2.2 節において述べたように、力を込める必要がなく、従来のタッチジェスチャと競合しない動作である。また追加のハードウェアを用いることもないため、高い汎用性を有している。

シェイク動作の検出には端末に組み込まれた加速度センサを用いた。具体的には、端末加速度の変化

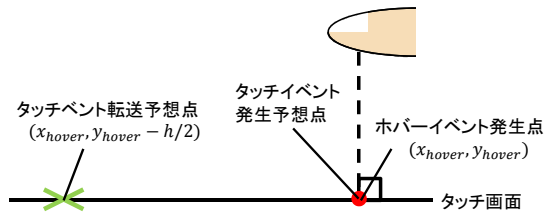


図 3. ホバーイベント発生点、タッチイベント発生予想点およびタッチ点転送予想点の座標関係。

が図 4 に示すように 600 ms 以上の間閾値を超えた際にシェイクが生じたとし、その後端末加速度の変化が閾値を下回った際に間接/直接操作を切り替えることとした。これにより、複数回の連続したシェイクを 1 回のシェイク動作として認識することができる。なおこの閾値は、シェイクを連続して行っている最中に加速度の変化が下回らない値を実験的に求め、本実装においては $200 \text{ m/s}^2/\text{s}$ とした。また切替の際、図 5 に示すようにメッセージとして「indirect/direct」を表示することによって現在の操作をユーザに示すこととした。

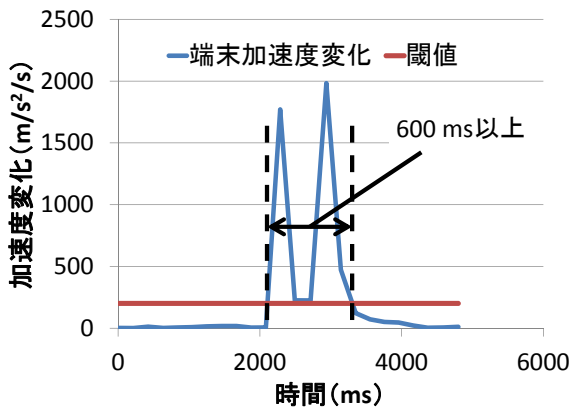


図 4. 端末をシェイクした際の加速度変化の様子。

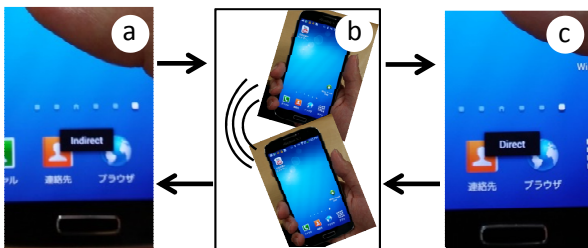


図 5. シェイク動作による TouchOver の起動/終了。
a: TouchOver の起動. b: 端末のシェイク動作.
c: TouchOver の終了。



図 6. 動画サイトへのアクセスおよび動画の再生. a: 指が直接届かない領域に表示された動画サイトのアイコン. b: シェイク動作による TouchOver の起動. c: TouchOver を用いたアイコンのタップ. d: TouchOver を用いた再生ボタンのタップ. e: TouchOver を用いたシークバーの操作。

4 使用例

本節においては、Galaxy S4 SC-04E (Android 4.2.2, $137 \times 70 \times 8 \text{ mm}$, $1920 \text{ px} \times 1080 \text{ px}$) にて実装したプロトタイプの使用例を述べる。なお、この端末はホバー検出が可能な端末であり、操作中使用者は利き手のみにて端末を把持した。

4.1 使用例 1: 動画サイトへのアクセスおよび動画の再生

TouchOver を用いた動画サイトへのアクセスおよび動画の再生手順を図 6 に示す。大端末のホーム画面に表示された動画サイトのアイコンに対して、ユーザの指が直接届かない場合を考える (図 6a)。この時ユーザは、シェイク動作を用いて TouchOver を起動する (図 6b)。画面にはホバーイベントを用いたタッチイベント転送先のフィードバックが緑色のカーソルとして表示されるため、このカーソルを左上のアイコンに合わせることでタッチ点の転送先を決定する。この状態にて画面をタップすることによってアイコンをタップする (図 6c)。同様にカーソルを画面上部の再生ボタンに合わせてタップすることにより、再生ボタンをタップする (図 6d)。カーソルを再生画面下部に合わせてドラッグ操作を行うことにより、シークバーを操作する (図 6e)。

この間手は大端末下部を安定に把持したままであり、大端末を楽に操作しているといえる。また、現在の動画を再生し終わり画面下部に表示されている別の動画にアクセスする際は、再び大端末をシェイクすることによって TouchOver を終了させ、直接リンクをタップすることによって次の動画にアクセスすることが可能である。

4.2 使用例 2: ダブルタップを用いたブラウザの拡大

ブラウザを用いて情報を確認している際、1 度の表示量を多くするために文字等の情報が縮小された状態にて表示される場合がある (図 7a)。ブラウザ

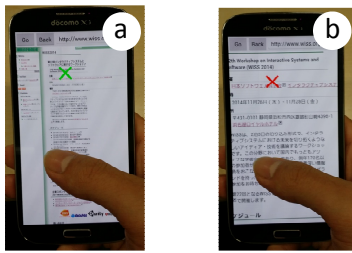


図 7. ダブルタップを用いたブラウザの拡大. a : 画面上部に小さく表示された情報. b : TouchOver を用いたダブルタップを行うことによる情報の拡大.

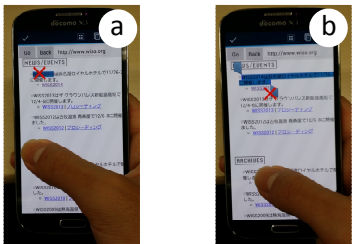


図 8. ロングタップを用いた文字列選択. a : TouchOver を用いてロングタップを行うことによる文字選択モードの起動. b : TouchOver を用いたドラッグ操作による選択ハンドルの移動.

上においてはダブルタップイベントが発生した座標を中心に画面が拡大されるが、今回目的とする主催情報は画面上部にあるため指で直接ダブルタップを行うことができず、また、この情報はウェブページの最上段に記述されているため、画面のスクロールを用いて手元に移動させることもできない。そこで、図 7b に示すようにカーソルを主催情報に合わせてダブルタップを行うことにより、カーソルの位置にダブルタップイベントが転送されるため、画面下部にて行ったダブルタップによって画面上部の拡大が可能となる。

4.3 使用例 3 : ロングタップを用いた文字列選択

ブラウザに表示された情報をコピーするため、表示されている文字列を選択する場合がある。ブラウザ上における文字列選択は、画面に対するロングタップを行い、表示された文字列選択ハンドルの範囲をドラッグ操作を行い調節することによって行う。今回は「WISS2014は浜名湖ロイヤルホテルで11/26-28に開催します。」という文字列を対象として選択操作を行う。しかしこの文字列は画面上部に表示されており、直接指にてロングタップを行うのは困難である。そこで図 8a に示すように、まず TouchOver を用いて先頭単語である「WISS2014」に対してロングタップを行い文字列選択モードを起動する。その後表示される文字列選択ハンドルの右側に TouchOver を用いてタッチを行い、続くドラッグ操作にてハン

ドルを文末まで移動させる(図 8b)。これにより、目的の文字列を選択することができる。

5 考察及び今後の課題

使用例 1-3 を考察する。本提案手法を用いることにより、従来の直接操作使用時に比べてはるかに楽に片手操作を行うことが可能となった。片手親指のみを用いて画面の全領域の操作を行うことができるため、シークバーなどの操作インタフェースが画面上部に配置されている YouTube などの動画サイトにおいては、その有用性が特に顕著であった。また、Wikipedia などの、文字の大きさが小さく頻繁な拡大縮小を必要とするウェブサイトにおいても、画面上部の拡大動作が非常に容易であり効率よく調べ物を行うことが可能であった。

一方、操作を行う際に、わずかではあるが持ち替えの必要性が生じた。これは親指が画面左端に対して届かないために生じたものである。この問題に対しては、タッチ点転送の方向を上方向のみから、たとえば上、左、左上の 3 種に拡張することによって改善が可能であると考えられる。これは Ninja Cursors [9] と同様にアイコンとカーソルの平均距離を縮める効果があるため、より小さな動作にて大端末の操作が可能になる。また、切り替え動作についても問題点が判明した。第 1 に、シェイク動作を行う際は画面を見ることができないため、直接/間接操作を切り替えたのちに操作対象を再び探す必要が生じる。第 2 に、大端末は重量があるため、連続して操作を切り替えると手が疲れるという問題が判明した。従って、シェイク以外のより軽量の切り替え動作を今後検討する予定である。これには現在、視線や操作時系列の状況等のコンテキストを用いることを考えている。

カーソルを用いたフィードバックについては、タッチ点転送予想点のフィードバックとしての働きに加え、現在 TouchOver が起動しているか否かの状態を示すフィードバックとしての有用性も感じた。また本手法は間接操作であるため、Shift などと同様に、操作に用いる指によって操作対象が隠れないという利点も確認された。一方でホバーイベントによるフィードバックは実際にタッチ点が転送される位置とわずかながらずれが確認された。これは、人間の指はホバー状態からタッチダウンを行う際、画面に対する厳密な垂直移動を行っていないため、ホバーイベントから予想されたタッチイベントの転送予想点と実際にタッチイベントが転送される点がずれるからであると考えられる。このずれに関しては被験者実験によりずれの量を定量的に評価し、フィードバックの表示位置を補正することにより解消される可能性がある。なお、ホバーイベントを用いたフィードバックがない状態における TouchOver の使用感を調査するための予備実験を行った結果、アイコン

程度の大きさの操作対象に対しては直接指を用いて操作するのと同程度の精度が確認された。したがって本手法は、たとえばホーム画面のアイコン操作等において、ホバー検出不可能な端末においても有用な手法であると考えられる。

6 まとめ

大端末向けの片手操作手法である TouchOver を示した。これは大端末を片手のみにて把持した際の操作を楽に行う、すなわち片手親指のみを用いた持ち替えを必要としない、画面の全領域に対する操作を行うための手法であり、画面下部にて発生したタッチイベントを画面上部に対して転送することによって実現される。このときタッチ点の転送先のフィールドバックとしてホバーイベントを用いたカーソルの表示を行い、TouchOver を起動/終了するための動作として大端末のシェイク動作を用いた。また、TouchOver を Galaxy S4 SC-04E 上に動作するプロトタイプアプリケーションとして実装し、その使用例を述べた。

今後は転送先の拡張や新たな切り替え動作を検討するとともに、ホバーによるフィールドバックの改良を行う。また、改良した TouchOver を用いて被験者実験を行い、実際に片手操作が楽に行えるかを、操作精度に関する定量評価や使用感に関するアンケートを用いた定性評価にて検証する。

参考文献

- [1] P. Baudisch, E. Cutrell, D. Robbins, M. Czerwinski, P. Tandler, B. Bederson, and A. Zierlinger. Drag-and-Pop and Drag-and-Pick: Techniques for Accessing Remote Screen Content on Touch- and Pen-operated systems. In *Proc of Interact Conference*, pp. 57–64, 2003.
- [2] A. Bezerianos and R. Balakrishnan. The Vacuum: Facilitating the Manipulation of Distant Objects. In *Proc of CHI '05*, pp. 361–370. 2005.
- [3] M. Goel, J. Wobbrock, and S. Patel. GripSense: Using Built-in Sensors to Detect Hand Posture and Pressure on Commodity Mobile Phones. In *Proc. of UIST '12*, pp. 545–554. 2012.
- [4] T. J. Gunn, H. Zhang, E. Mak, and P. Irani. An Evaluation of One-handed Techniques for Multiple-target Selection. In *CHI '09 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '09, pp. 4189–4194, 2009.
- [5] C. Harrison, J. Schwarz, and S. E. Hudson. TapSense: Enhancing Finger Interaction on

- Touch Surfaces. In *Proc. of UIST '11*, pp. 627–636. 2011.
- [6] S. Heo and G. Lee. Force Gestures: Augmented Touch Screen Gestures Using Normal and Tangential Force. In *Proc. of CHI EA '11*, pp. 1909–1914. 2011.
- [7] S. Heo and G. Lee. Forcetap: Extending the Input Vocabulary of Mobile Touch Screens by Adding Tap Gestures. In *Proc. of MobileHCI '11*, pp. 113–122. 2011.
- [8] A. K. Karlson and B. B. Bederson. One-handed Touchscreen Input for Legacy Applications. In *Proc. of CHI '08*, pp. 1399–1408. 2008.
- [9] M. Kobayashi and T. Igarashi. Ninja Cursors: Using Multiple Cursors to Assist Target Acquisition on Large Screens. In *Proc. of CHI '08*, pp. 949–958, 2008.
- [10] J. Lai and D. Zhang. ExtendedThumb: A Motion-based Virtual Thumb for Improving One-handed Target Acquisition on Touch-screen Mobile Devices. In *Proc of CHI EA '14*, pp. 1825–1830. 2014.
- [11] P. Parhi, A. K. Karlson, and B. B. Bederson. Target Size Study for One-handed Thumb Use on Small Touchscreen Devices. In *Proc. of MobileHCI '06*, pp. 203–210. 2006.
- [12] E. W. Pedersen and K. Hornbæk. Expressive Touch: Studying Tapping Force on Tabletops. In *Proc. of CHI '14*, pp. 421–430. 2014.
- [13] V. Roth and T. Turner. Bezel Swipe: Conflict-free Scrolling and Multiple Selection on Mobile Touch Screen Devices. In *Proc. of CHI '09*, pp. 1523–1526. 2009.
- [14] A. Roudaut, S. Huot, and E. Lecolinet. TapTap and MagStick: Improving One-handed Target Acquisition on Small Touch-screens. In *Proc. of AVI '08*, pp. 146–153. 2008.
- [15] D. Vogel and P. Baudisch. Shift: A Technique for Operating Pen-based Interfaces Using Touch. In *Proc. of CHI '07*, pp. 657–666. 2007.
- [16] Y. Wu, T. Mei, N. Yu, and S. Li. Accelerometer-based Single-handed Video Browsing on Mobile Devices: Design and User Studies. In *Proc of ICIMCS '12*, pp. 157–160. 2012.
- [17] N.-H. Yu, D.-Y. Huang, J.-J. Hsu, and Y.-P. Hung. Rapid Selection of Hard-to-access Targets by Thumb on Mobile Touch-screens. In *Proc. of MobileHCI '13*, pp. 400–403. 2013.
- [18] 土佐伸一郎, 田中二郎. LoopTouch: 画面ループを用いたモバイル端末片手操作手法. インタラクシオン 2013, 8 pages, 2013.