

腕時計型端末におけるズーム・スクロールを用いたタッチ入力

原清貴^{†1} 梅澤猛^{†2} 大澤範高^{†2}

タッチパネルを搭載した腕時計型端末は携帯性と操作性に優れるが、ディスプレイサイズの制約が強いため、表示する情報量を増やすと操作が困難になるという問題がある。そこで本研究では、ズームとスクロールを併用することで狭い領域で効率的な操作を実現するタッチ入力法を提案する。スクロールの変位量に応じた自動ズームにより、一度の操作でズームとスクロールの両方を行う。提案手法を用いたアプリケーションをタブレット端末上に実装したところ、28mm × 28mm の入力領域で良好な文字入力操作が可能であることが確認できた。

Touch Input Method with Zooming and Scrolling for Smartwatches

KIYOTAKA HARA^{†1} TAKESHI UMEZAWA^{†2} NORITAKA OSAWA^{†2}

A smartwatch has a touchscreen for input and output. However touch user interfaces on smartwatches need a new input method, because the size is much smaller than smartphone's display size. In this study, we propose a touch input method using zooming and scrolling. And we consider automatic zooming based on displacement of scrolling to reduce the number of operations. We developed a prototype text entry system in the input area whose size is 28 mm × 28 mm.

1. はじめに

SONY の SmartWatch や Samsung の GALAXY Gear のように、タッチパネルを有する腕時計型端末が新たな情報通信端末の 1 つとして注目されている。腕時計型端末は小型で身に着けやすいため携帯性に優れている。また、スマートフォンなどと同じように操作をタッチパネルで行えるため、新たに操作法を覚える手間が少ない。しかし、ディスプレイサイズの制約が強いため、表示する情報量を増やすと操作が困難になるという問題がある。

タッチパネルインタフェースでは、ディスプレイに表示されたウィジェットを指で操作する。しかし、腕時計型端末で多数のウィジェットを表示すると、サイズは指より小さくなる。小さなウィジェットを指で操作する場合、それぞれの区分を視覚的に認識できたとしても押し分けが困難という“fat fingers[1]”の問題が発生するため、正確な操作が難しくなる。また、松浦らはスマートフォンを片手で把持し、同じ手の親指でタップする場合には、ウィジェットのサイズを 7 mm × 7 mm 以上にすることが望ましいとしている[2]。現在の腕時計型端末では、配置するウィジェットの数を少なくすることで入力に必要なサイズを確保しているため、扱える入力情報が少なくなり、実装可能な機能を制限している。

そこで本研究では、ズームとスクロールを併用することで狭い領域での効率的な操作を実現するタッチ入力法を提案する。提案手法によりディスプレイ上で利用可能なウィ

ジェット数を増やすため、扱える入力情報が多くなる。これは腕時計型端末の機能を拡張する役に立つと考える。提案手法は、ズーム操作によるディスプレイ上に表示されるウィジェットのサイズと数の調整と、スクロール操作によるズーム位置の修正を可能にすることで実現する。また、使いやすさを考慮すると入力に必要な時間が短いことが望ましい。そこでスクロールに合わせてズーム率を変更する方法を提案し、入力に必要な操作数を減らし、入力時間を短くすることを目指す。

2. 関連研究

ズーミングインタフェースに関する研究は多く、タッチパネル上で小さなウィジェットへのタッチ入力をズームにより可能にする方法も提案されている。しかし、腕時計型端末ほど小さなディスプレイサイズに対応できるものは多くない。

Olwal らはサイズの大きいシングルタッチ式ディスプレイを対象に、擦ったターゲットの拡大率を擦る操作の向きによって調整する手法を紹介している[3]。この擦るという動作は指 1 本で行えるため、狭い入力領域でも利用できる。しかし、フリックなど擦る動作に似た操作がすでに利用され、それらのジェスチャによって実行される操作のイメージが出来上がっているため、動作が似ているジェスチャに対してイメージに合わない操作を定義すると、操作方法を覚える手間がユーザの負担となる。また、ボタンへのタップのようにウィジェットへの入力は指 1 本で行うので、操作の誤検知を防ぐためにズームとスクロールには 2 本の指ジェスチャを利用することがよいと考える。

^{†1} 千葉大学工学部情報画像学科

Department of Informatics and Imaging Systems, Faculty of Engineering, Chiba University

^{†2} 千葉大学大学院融合科学研究科

Graduate School of Advanced Integration Science, Chiba University

Oney らは超小型のタッチパネル上での文字入力法として ZoomBoard を開発した[4]. ZoomBoard は, QWERTY 配列のソフトウェアキーボードを利用している. ソフトウェアキーボード全体が表示されている状態 (入力初期状態) でタップすると, タップしたキー周辺が拡大される. ダブルタップすると, タップした場合より拡大される. 拡大後にキーをタップすることで入力する. 入力後は, 入力初期状態に自動で戻る. そのため, 1 文字入力するには 2 回以上のタップが必要になる, ズーム位置やズーム率を修正する場合, 入力初期状態に戻りズーム操作をやり直す必要があるなどユーザビリティに課題がある.

スクロールに合わせてズーム率を変更する手法は五十嵐らがスクロールの移動速度に応じた自動ズームを提案している[5]. スクロール速度が速くなると縮小表示した画面になり, 速度が遅くなると拡大表示した画面になる. しかし, これは PC 向けに開発されたものであり, 腕時計型端末の操作領域は狭いため, 自動ズームの基準として利用できるほどにスクロールの速度に差をつけることが難しく, 腕時計型端末で必要とされる移動距離も一般的には PC よりも小さい.

3. 提案手法

本研究では, 腕時計型端末のディスプレイのように狭い入力領域でも使用可能なズームとスクロールを利用した入力法を提案する. 提案手法はソフトウェアキーボードを対象としたものである. 以下では, ズームとスクロールのそれぞれをタッチジェスチャで行う場合とスクロールの変位量に応じて自動でズームする場合の 2 つの提案手法を説明する.

3.1 ズームとスクロールを 2 指によるジェスチャで定義

キー入力是指 1 本によるタップ, ズーム操作とスクロール操作をそれぞれ指 2 本によるタッチジェスチャで定義することで, キー入力とズーム・スクロール操作を明確に分ける. 指の数で操作を区別することで, 操作の誤検知を防ぎ, ズームの修正を容易にする. ズーム操作とスクロール操作の図解例を図 1 に示す.

ズーム操作には, ズーム操作には, 図 1 (b), (c) に示すようなピンチイン・ピンチアウトに似たジェスチャを用いる. 通常, ピンチイン・ピンチアウトによるズーム操作では, 2 指の移動量を加味した中間の位置をズームの中心とし, 2 本の指をそれぞれ動かすことで拡大を調整する. しかし, 想定する入力領域では 2 本の指のそれぞれを動かせるほどのスペースを用意できないため, 1 本の指 (固定指) をほとんど動かさず, もう 1 本の指 (動作指) のみ動かすジェスチャを主たる利用方法と考える. このジェスチャでは, 固定指でズーム位置を指定すると操作がわかりやすいと考え, 固定指付近をズームの中心とし, 動作指で拡

大を調整するようにズームを実装する. ズームが不十分な場合には動かしていた方の指を画面から離して別の位置からジェスチャを再開することで, ズーム操作を続けることができる. スクロール操作には, 図 1 (d) のように 2 本指スクロールを用いる.

ズームとスクロール操作に用いられるジェスチャは, 操作を覚える手間が少なくするために, スマートフォンなどで使われているものと関連付けやすいものを採用した. また, ジェスチャを組み合わせることで, ズームとスクロールを同時に行うこともできる.

画面右向きに X 軸, 上向きに Y 軸がある座標系において, 時刻 t のある点 A の座標を (x, y) , その直前の時刻 t' の点 A の座標を (x', y') とし, 時刻 t の指の位置をそれぞれ p, q , 時刻 t' の指の位置をそれぞれ p', q' とする.

画像の拡大縮小では原点を中心として拡大縮小するので, 任意の点をズームの中心として拡大縮小する場合は

- ① ズームの中心を原点へ移動する
- ② 画像を拡大, 縮小する
- ③ ズームの中心を元の位置へ移動する

という処理が必要となる. ズームの中心の座標を (x_c, y_c) , ズーム率を s とすると, この一連の処理は以下の式で示される.

$$\textcircled{1} \begin{pmatrix} x_1 \\ y_1 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & -x_c \\ 0 & 1 & -y_c \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} \quad (1)$$

$$\textcircled{2} \begin{pmatrix} x_2 \\ y_2 \\ 1 \end{pmatrix} = \begin{pmatrix} s & 0 & 0 \\ 0 & s & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ y_1 \\ 1 \end{pmatrix} \quad (2)$$

$$\textcircled{3} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & x_c \\ 0 & 1 & y_c \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_2 \\ y_2 \\ 1 \end{pmatrix} \quad (3)$$

(1), (2), (3) 式をまとめるとズームは以下のように示される.

$$\begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{pmatrix} s & 0 & (1-s)x_c \\ 0 & s & (1-s)y_c \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} \quad (4)$$

$$s = c(\|p - q\| - \|p' - q'\|) \quad (5)$$

$$x_c = \min(p_x, q_x) + \frac{|\Delta p_x|}{|\Delta p_x| + |\Delta q_x|} \quad (6)$$

$$y_c = \min(p_y, q_y) + \frac{|\Delta p_y|}{|\Delta p_y| + |\Delta q_y|} \quad (7)$$

c は定数を示す. $\|v\|$ はベクトルの長さ, $\min(a, b)$ は引数の最小値と定義する. v_x はベクトルの x 成分, v_y はベクトルの y 成分, Δv_x は時刻 t' から時刻 t までのベクトルの x 成分の変位量 ($v_x - v'_x$), Δv_y は時刻 t' から時刻 t までのベクトルの y 成分の変位量 ($v_y - v'_y$) と定義する. (4), (5) 式によって, 指を広げると拡大, 狭めると縮小する. (6), (7) 式により 2 指の動かさない指の位置付近をズームの中心とする.

また, スクロールは以下の式で示される.

$$\begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & \min(\Delta p_x, \Delta q_x) \\ 0 & 1 & \min(\Delta p_y, \Delta q_y) \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} \quad (8)$$

(8) 式によって、ジェスチャに合わせて表示位置を動かすことができる。2 指のうち変位量が小さい指の変位量にあわせて表示位置は変化する。

ズームとスクロール同時に行った場合、ズームとスクロール同時に行った場合は以下のように (4), (8) 式をまとめた式で示される。

$$\begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{pmatrix} s & 0 & (1-s)x_c + \min(\Delta p_x, \Delta q_x) \\ 0 & s & (1-s)y_c + \min(\Delta p_y, \Delta q_y) \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} \quad (9)$$

s は (5) 式, x_c は (6) 式, y_c は (7) 式で定義される。

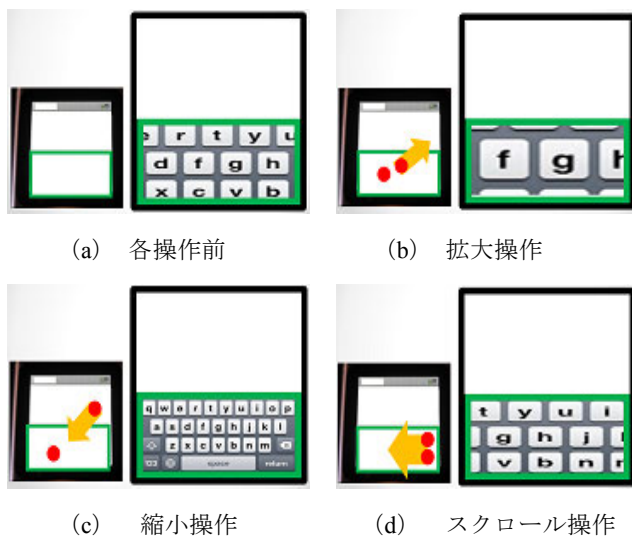


図 1 ズーム操作とスクロール操作

Figure 1 Zooming gesture and Scrolling gesture

3.2 スクロールの変位量に応じた自動ズーム

キー入力是指 1 本によるタップ, スクロール操作は 2 本指スクロール用いる。ズームはスクロールの変位量に合わせて自動で行われる。スクロールの変位量に応じた自動ズームのイメージを図 2 に示す。スクロールの変位量が小さいほど 2 本の指が触れている部分が拡大, スクロールの変位量が大いほど縮小される。自動でズーム率が調整されるため, 想定する入力領域のように狭い範囲でのスクロール操作でも操作領域以上の距離のスクロールを可能にする。また, スクロール操作を複数回繰り返すと変位量が大きくなるため, さらに縮小され, 長距離のスクロールを実現する。

画面左向きに X 軸, 上向きに Y 軸がある座標系において, ある時点での指の位置をそれぞれ p, q , その直前の指の位置をそれぞれ p', q' とする。スクロールの変位量 d に応じた自動ズームは式 (9) において, ズーム率 s を以下の式で定義することで実現する。

$$s = \frac{c'}{\min(d, n)} \quad (10)$$

$$d = \frac{\|p-p'\| + \|q-q'\|}{2} \quad (11)$$

c', n は定数, $\|v\|$ はベクトルの長さを示す。 $\min(a, b)$ は引数の最小値と定義する。(10) 式によって, スクロールの変位量が n 以下の場合はスクロールのみ, n より大きい場合は変位量に応じて縮小する。

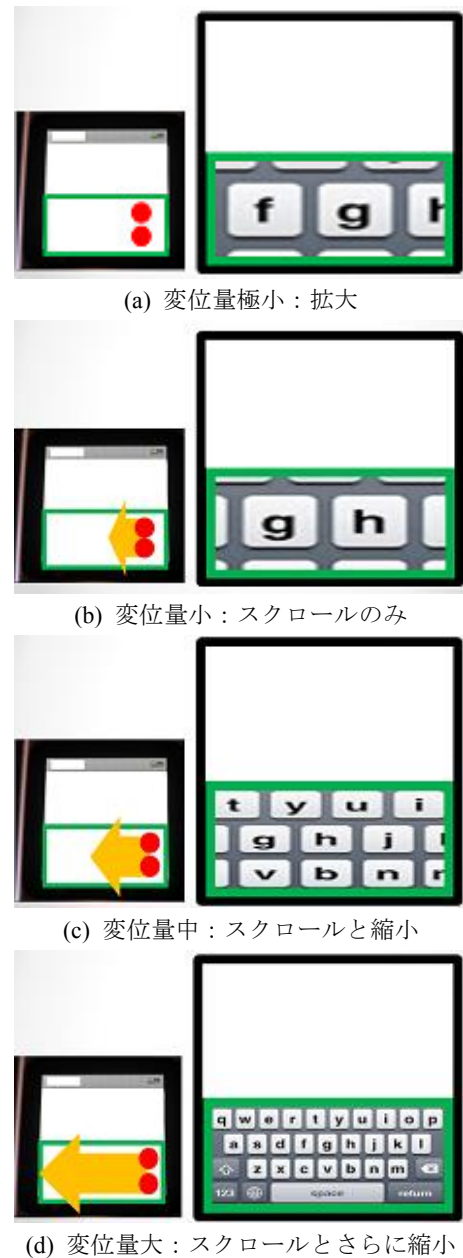


図 2 スクロールの変位量に応じた自動ズーム

Figure 2 Automatic zooming based on displacement of fingers for scrolling

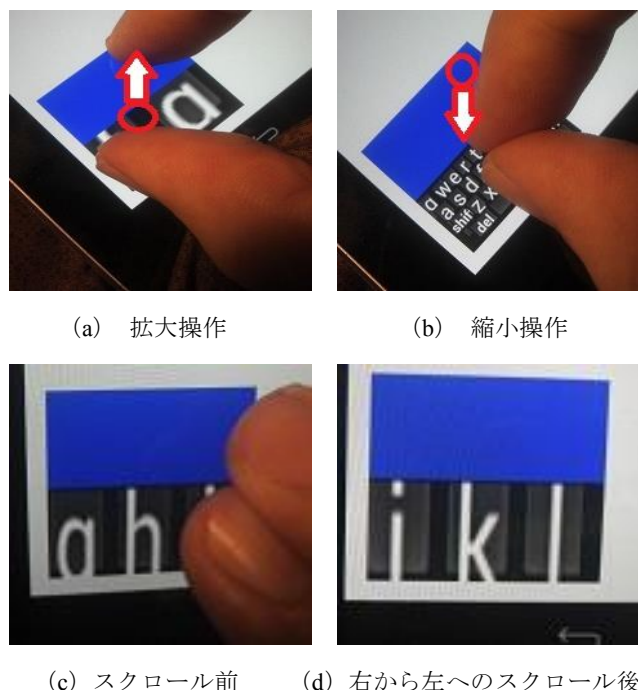
4. 実装

提案手法を評価するためにズームとスクロール機能を持ったソフトウェアキーボードを開発した。ソフトウェアキ

ーボードは Google Nexus7 (2012) (OS : Android 4.2.2) 上で動作するアプリケーションとして実装した。さまざまな大きさの腕時計型端末が存在するため、本研究ではディスプレイサイズが 1.54 インチである iPod nano (第 6 世代) を参考に、入力に利用する領域を 28 mm × 28 mm とした。また、ソフトウェアキーボードはディスプレイの下半分に表示されるとする。これは、上半分の領域が入力文字の表示などに使われることを考慮したためである。

図 3 に実装したソフトウェアキーボードの操作の様子を示す。このソフトウェアキーボードは、ズームとスクロールを別のジェスチャで定義したものである。青色の領域は、想定したディスプレイサイズの上半分を表している。現状では、ズーム操作による拡大、縮小とスクロール操作による表示位置の移動は実装している。ソフトウェアキーボードの縮小による最小サイズはキーボード全体がキーボード表示領域にちょうど収まる大きさとした。これは、キーボード全体が表示領域より小さくなる必要はないと考えるためである。

今後は、ソフトウェアキーボードが使いやすくなるように改良を続けていく必要がある。また、スクロールの変位量に応じた自動ズームの実装も進めていく予定である。



(a) 拡大操作 (b) 縮小操作

(c) スクロール前 (d) 右から左へのスクロール後

図 3 プロトタイプソフトウェアキーボード

Figure 3 Prototype software keyboard

5. おわりに

ズーム・スクロールを用いることで腕時計型端末のディスプレイサイズの操作領域で実現可能なタッチ入力を提案し、提案手法を利用した文字入力を実現した。今後は、被

験者による評価実験を行う予定である。評価実験では、提案手法として、ズーム・スクロールを別のジェスチャで定義したソフトウェアキーボード、スクロールの変位量に応じた自動ズームを実装したソフトウェアキーボードと、既存手法である ZoomBoard を再現したソフトウェアキーボードを比較することで評価する。評価実験のために作成したタイピングアプリを用いて入力速度、入力エラー率を評価し、質問紙調査により使いやすさを評価する。

参考文献

- 1) Siek, K. A., Rogers, Y. and Connelly, K. H.: Fat finger worries: How older and younger users physically interact with PDAs, *Proc. IFIP TC13 International Conference on Human-Computer Interaction*, pp.267-280 (2005).
- 2) 松浦吉祐, 郷健太郎: 小型タッチ画面における片手親指の操作特性, *ヒューマンインタフェース学会論文誌*, Vol.9, No.4, pp.37-44 (2007).
- 3) Olwal, A., Feiner, S. and Heyman, S.: Rubbing and Tapping for Precise and Rapid Selection on Touch-Screen Displays, *Proc. CHI 2008*, pp.295-304 (2008).
- 4) Oney, S., Harrison, C., Ogan, A. and Wiese, J.: ZoomBoard: A Diminutive QWERTY Soft Keyboard Using Iterative Zooming for Ultra-Small Devices, *Proc. CHI 2013*, pp.2799-2802 (2013).
- 5) 五十嵐健夫, Ken Hinckley: 移動速度に応じた自動ズームによる効率的ナビゲーション, *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(WISS 2000)*, pp.57-66 (2000).