

Bring Your Own Pointer : 複数の携帯端末による ad hoc なマルチポインティング手法

佐藤 光起^{1,a)} 松下 光範^{1,b)}

受付日 2020年4月17日, 採録日 2020年11月5日

概要: 本研究の目的は、実空間において複数のユーザが1つの共有画面に対して ad hoc に利用可能なポインティング手法の実現である。複数人によるポインタの同時操作では、利用人数の増加にともなってユーザが自身のポインタを見失いやすくなることや、ユーザの人数分のポインティングデバイスが必要になるといった問題がある。そこで本研究では、指定された URL にユーザが自らのスマートフォンでアクセスすることで、そのスマートフォンをポインティングデバイスとして利用可能にする手法 BYOP (Bring Your Own Pointer) を提案する。この手法では、スマートフォンに標準搭載されているセンサのみを用いてポインティング位置を判断している。本稿では、既存のジャイロマウスや光学マウスとの比較実験、ならびに ISO9241-9 に基づく性能評価実験を通じて、提案手法の有用性を確認した。

キーワード: インタラクション, ヒューマンインタフェース, 携帯端末, ジャイロセンサ

Bring Your Own Pointer: An ad hoc Pointing Method Using Multiple Smartphones

KOKI SATO^{1,a)} MITSUNORI MATSUSHITA^{1,b)}

Received: April 17, 2020, Accepted: November 5, 2020

Abstract: The aim of this study is to realize a pointing method that multiple people can use for ad hoc. Simultaneous operation by multiple users with pointers has the problem that a user loses his or her pointer at the number of users increases; further multiple operation terminals are required. In this study, we propose a pointing method called Bring Your Own Pointer (BYOP) that only uses a gyro sensor of a smartphone. Experiments to evaluate the performance of the pointing system based on ISO9241-9 and to identify a user's cursor from multiple cursors were conducted. It was confirmed that the proposed method is superior to existing methods as a pointing method that can be operated by multiple users simultaneously.

Keywords: interaction, human interface, mobile terminal, gyro sensor

1. はじめに

大型スクリーンを用いた対面同一場での議論やプレゼンテーションの場面においては、単一のスクリーンを共有しそれに複数人がポインティングするという行為がしばしば起こりうる。こうしたポインティングの場面ではレーザーポインタや指示棒がよく用いられるが、共有するスクリーン

に投影されている情報がプレゼンテーションツールによるものであったり計算機のミラーリング画面であったりする場合には、マウスやトラックパッドなどのポインティングデバイスもポインティング手段としてしばしば用いられる。こうした場面では、ポインティングデバイスが1つしかないためにそれを把持するユーザのみがポインティング操作を行い、ポインティングデバイスを把持していない他のユーザは、口頭で指し示すポイントを指示したり、提示される情報を一方的に視聴するだけになることがある。こうした場面では、唯一のポインティングデバイスを把持しているユーザ以外は画面上の位置を指示しにくいという問

¹ 関西大学
Kansai University, Takatsuki, Osaka 569-1095, Japan
a) k081485@kansai-u.ac.jp
b) mat@res.kuttc.kansai-u.ac.jp

題が発生し、議論やコミュニケーションの阻害要因となる。こうした問題を解決するため、複数のユーザによる同時ポインティングを可能にする手法が提案されている（たとえば文献 [3], [19]）。これらの手法では、各ユーザに対応付けられたポインタがディスプレイ上に複数表示され、ユーザはそのポインタを介して操作を行う。しかし、人数分のポインティングデバイスを事前に用意する必要が生じることや、ad hocに議論やプレゼンテーションに参加することが想定されていない問題がある。

また、複数のユーザがポインタを同時に操作する手法では、同一画面上にユーザの人数に応じた数のポインタが表示されるため、同時に表示されるポインタの数が増加するとユーザは自身のポインタを認識することができなくなり、結果として操作が困難になる懸念がある。これはポインタ数の増加に応じてより深刻なものとなり、ユーザが自身のポインタを認識するまでにかかる時間の増加を招く [13]。この問題については、ユーザごとにポインタの色や形を変えて固有にすることで識別性を向上させて解決する試みが行われているが（たとえば文献 [16], [17]）、ポインタの外見が固定されるため、自身が描いた絵をポインタアイコンとして利用するような、ポインタをインタラクティブに変化させるコンテンツに対する制約（たとえば、ポインタとして利用可能な色の制限）が生じる懸念がある。

そこで本研究では、ユーザが「スマートフォンを把持した手を伸ばし、画面上の点を直接指し示す」というポインティング行動を採用することによってユーザの動作とポインタの座標との関連を強め、多数のポインタが同時に表示される場合においても短時間で自身のポインタを直感的に認識可能にすることで、このポインタを見失う問題の解決を図る。加えて、自らが所有するスマートフォンをポインティングデバイスとして利用することで、事前に多数のポインティングデバイスを用意しなくてはならない問題と、議論やコミュニケーションへの ad hoc な参加の問題についても同時に解決する。

本稿では、こうした協同ポインティング手法としてBYOP (Bring Your Own Pointer) を提案する (図 1 参照)。提案手法ではポインタの形状や色彩を変化させることなく、ユーザに自身のポインタを認識させる。そのためポインタで指し示す座標でのみコミュニケーションが行われていた従来手法に比べ、ポインタの色や形を活用した情報交換を実現可能である。

2. 関連研究

2.1 ポインティング手法の分類

ポインタの操作手法は、主に単一ユーザが操作することを想定して広く研究が行われている。ここではそうしたポインタの手法を、Sweetser らの分類 [10] を参考に分類を行い、それぞれの利点と欠点について述べる。ポインティ

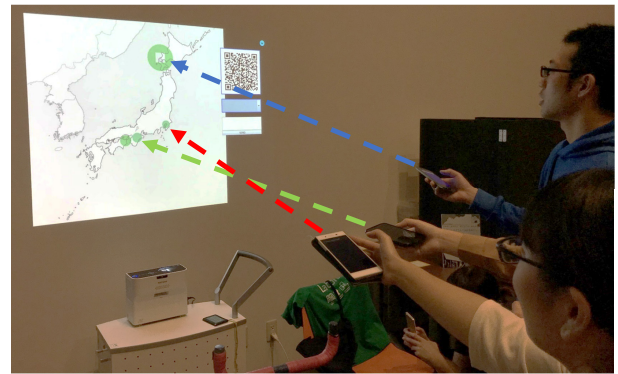


図 1 提案手法 BYOP の使用例. スマートフォンのみを用いてポインタを操作可能

Fig. 1 Example of the BYOP method. The proposed method permits a user to operate the pointer using only a smartphone.

ング手法はユーザが操作する端末やユーザ自身の身体位置と、ポインティングが行われるディスプレイ上の位置関係に着目することで、相対ポインティング手法と絶対ポインティング手法の2つに分類できる。

2.1.1 相対ポインティング手法

相対ポインティング手法は、ユーザの手元の操作量を計測することでポインタを操作する手法である。代表的な相対ポインティング機器としてマウスがあり、端末移動距離に応じてポインタを操作できる。マウスには一般的な光学式の端末以外に、ジャイロセンサを活用した端末（ジャイロマウス）がある。ジャイロマウスは端末に搭載されたジャイロセンサによって手元の回転角を計測し、それに従ってポインタが移動する。

宮田らは携帯端末上をスワイプすることで、ネットワークを介して接続されているディスプレイ上のポインタを操作できる手法を開発した [17]。瀧口らは、カメラ画像を活用し、事前に登録した単色の矩形物体（カード）をマーカとしてポインタ操作を実現している [18]。また、Babic らの Pocket6 はユーザの手元に仮想ディスプレイを設定しスマートフォンの加速度センサと深度センサの値から仮想ディスプレイ上の端末位置を計測し、その値をもとにポインタを表示する座標を決定する [1]。加速度センサを活用した手法は、画像認識を活用した手法に比べ計算量が少なく高速なポインティングを可能にする。

これらの相対ポインティング手法は端末の移動量のみを活用して操作するため、操作端末とディスプレイの空間上の位置関係に拘束されず、ユーザの位置や体勢にかかわらず利用できる。一方、ユーザの向きとポインタの位置に関係がないため自身のポインタの位置を見失いやすいことや、ディスプレイが大型化した場合に手元での操作量が増加する問題がある。

2.1.2 絶対ポインティング手法

絶対ポインティング手法では入力値が示した位置にポイ

ンティングを行う。この手法では、ユーザが指や端末で示した位置や触れた位置へポインティングが行われる。

Wilson らの XWand は、ユーザが持つ端末から照射した赤外光と、ジャイロセンサの値を活用し三次元空間においてユーザが示している方向を算出できる [11]。新谷らはユーザが指差しているディスプレイの座標をカメラ画像から算出することでポインタを操作する手法を提案している [15]。牛田らは、操作者（ユーザ）が持つペンライトの光をカメラで観測し、操作者が指し示していると思われる画面上の座標を算出して、ディスプレイ上の該当位置にポインタを表示する手法を提案している [12]。こうした専用端末を用いた操作手法は正確である一方、複数のユーザが同時に操作を行う場合はユーザの数に応じた端末が必要になり、機器の準備に負担がかかるという問題がある。

天早らの Remote Touch Panel は Kinect によってユーザの姿勢を計測し、その手の位置から指し示している座標を算出する [14]。身体トラッキングによって操作を行う手法では、ユーザの数が増加しても専用機器が必要でない利点があるが、トラッキングの精度が環境的要因によって大きく左右する。

Pan らの EasyPointer はスマートフォンのジャイロセンサを用いてポインティングを実現している [7]。これはユーザがスマートフォンをディスプレイ上下左右の 4 隅へ向けた際のジャイロセンサ値を計測し、それをもとにキャリブレーションを行う。キャリブレーションによって操作端末とディスプレイの位置関係を決定することで、絶対ポインティングが可能となる。

絶対ポインティング手法はユーザが指し示した先にポインタが表示されるため、直感的なポインティングを実現する。しかし、ポインティングシステムがユーザとディスプレイの位置関係を検出する必要があり、どのように位置関係の入力を行うかが問題となる。

2.2 複数のユーザが同時に利用可能なコンピュータ操作手法

柴崎らは博物館のミュージアムシアタにおいて、ユーザ各々が操作可能なポインタを複数ディスプレイに表示することで、ユーザ間で気づきの共有や自身の興味の表出などのコミュニケーションが行われることを報告した [19]。同時に複数のポインタを表示することでユーザが自身のポインタを見失う問題については、渡邊らによってユーザが認識可能であることが示唆されていた [13]。しかし柴崎らがミュージアムシアタで行った実験では、ポインタを見失った被験者がいたことが報告されている。このことから、実際にコンテンツを操作する環境下において、一定数のユーザはポインタが増加することで自身のポインタを認識できなくなり、操作が不可能になることがあると考えられる。

牟田らはユーザが操作可能なポインタを色で示すこと

で、よりユーザが自身のポインタを認識しやすい手法を提案し、複数人によるポインタの同時操作が新たなコンテンツとしてユーザの注目を喚起することや、ユーザにとって楽しめる体験を提供することを示唆した [16]。宮田らはポインタの形状をユーザごとに固有にすることで視認性が向上することを報告した [17]。しかし、これらの手法では同時に表示されるポインタの数が増加した場合ユーザが自身のポインタを認識する精度が低下することが課題となっている。また、これらポインタの外見を変化させることで認識を補助する手法では、コンテンツに対して制約を与えることになる。この問題は特にエンタテインメントなどへの活用を考える上で大きな課題となる。

3. デザイン指針

本研究は複数人が同時に利用できるポインティング手法の提案を目的としている。複数のユーザにより利用される手法は ad hoc に利用できることが求められるが、ポインタの操作を目的とした手法では、同時にポインタを操作する人数が増加した場合、自身のポインタを見失い操作ができなくなる問題や、ポインタを発見するためにかかる時間が増加する問題があり [13], [16]、これらの問題は ad hoc な利用を妨げると想定される。これら問題は特に、ユーザの動作とポインティングを行う位置とに関連のない相対ポインティング手法ではその傾向が顕著であると考えられる。一方、絶対ポインティング手法はユーザの動作とポインティングされる位置との関連が強くユーザが容易にポインタを認識できるが、専用端末による操作は利用人数の増加にともない必要な装置が増加することが課題となる。

そこで、本システムが満たすべき要件を、

- (1) ポインティング手法として使用性が十分であること、
 - (2) ユーザが容易に自身のポインタを認識できること、
 - (3) ユーザが ad hoc に利用できること、
 - (4) ユーザが特殊な機器を準備する必要がないこと、
- の 4 点とする。

4. 実装

3 章で定めた要件を基に、システムを実装する。要件 (2) は、絶対ポインティング手法を用いることで実現する。要件 (3) について、ad hoc な利用を妨げている主たる理由はユーザがポインタを見失うことであるため、要件 (2) を満たすことで同時に実現しているが、多人数が利用するシステムにおいてユーザが容易に利用可能であることが重要と考えられる。そこで、ユーザが QR コードを読み込むだけで利用可能なシステムを提案する。機器のペアリングなどの操作はユーザ数が増加するにつれその手間が問題になる。本手法では、設定を容易にすることでそうした問題の解決を図る。要件 (4) は、スマートフォンに搭載されているジャイロセンサおよび地磁気センサの値を用いてポイン

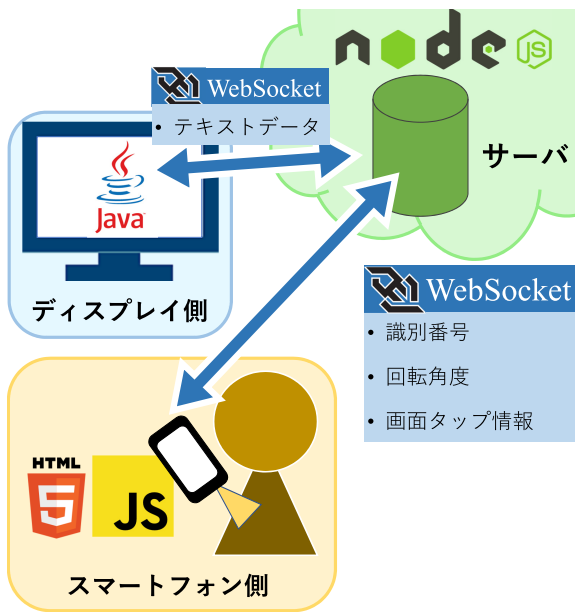


図 2 BYOP システム構成図
Fig. 2 The system configuration of the BYOP.

タを操作する手法を実現することで達成する。

4.1 システム構成

提案手法のシステム構成図を図 2 に示す。提案手法はスマートフォン側アプリケーション、ディスプレイ側アプリケーション、外部サーバで構成されている。各アプリケーションは WebSocket によって接続される。

4.1.1 スマートフォン側アプリケーション

コントローラとなる Web アプリケーションを HTML5 と JavaScript を用いて実装した。このコントローラアプリケーションはサーバを介してディスプレイ側アプリケーションへ情報を送信する。送信する項目を以下に示す。

- (1) コントローラアプリケーションごとに固有の識別値
- (2) スマートフォンの α , β , γ 回転角
- (3) スマートフォン上のタップされた X, Y 座標
- (4) 画面をタップされているか否か
- (5) 画面をタップされた回数
- (6) その他機能拡張に必要な情報 (メッセージなど)

スマートフォンの回転角は JavaScript の Device Orientation イベント*1を用いて取得する。これはスマートフォンに搭載されているジャイロセンサや地磁気センサの値をもとに、図 3 のように定義される。z, x, y を軸としたときの Euler 角 α (α), β (β), γ (γ) の変化を検知し、その値を取得できるものである。座標系は、スマートフォンの画面を横方向に貫く軸を x 軸、画面を縦方向に貫く軸を y 軸、画面に垂直な軸を z 軸と定義される。 α は z 軸周りの回転角度であり、スマートフォンの左右方向の回転を取得できる (図 3(a) 参照)。回転角 α は 0 から 360 の値で表現され、Device Orientation イベントが初期

化されたときのスマートフォンの向きにおける値は 0 である。以降、スマートフォンを左方向に回転させると値が増加する。 β は x 軸周りの回転角度であり、スマートフォンの前後方向の傾きを取得できる (図 3(b) 参照)。回転角 β は -180 から 180 の値で表現され、Device Orientation イベントが初期化されたときのスマートフォンの向きにおける値は 0 である。以降、スマートフォンを後方向に傾けると値が増加し、前方向に傾けると値が減少する。 γ は z 軸周りの回転角度であり、スマートフォンの左右方向の傾きを取得できる (図 3(c) 参照)。回転角 γ は -90 から 90 の値で表現され、Device Orientation イベントが初期化されたときのスマートフォンの向きにおける値は 0 である。以降、スマートフォンを右方向に傾けると値が増加し、左方向に傾けると値が減少する。

4.1.2 ディスプレイ側アプリケーション

ディスプレイで使用するアプリケーションを Java SE Development Kit 9*2で実装した。このアプリケーションとスマートフォン側アプリケーションはユーザがコントローラページにアクセスすることで、サーバを介して通信を開始する。その後、第 1 点をディスプレイ上端右端、第 2 点をディスプレイ下端左端として、ユーザがスマートフォンによりこれら 2 つの基準点を指し示したときの回転角を登録する。

本システムはスマートフォンの回転角を基に座標を決定するため、ユーザの正面においてその値が連続する必要がある。取得した値のうち α は初期化された時点の値が 0 であるが、その際の端末の方向からさらに右方向に端末を回転させると α は 360 となり不連続である。そのためセンサから取得される α の値を α_{raw} として式 (1) のように調整する。

$$\alpha = \begin{cases} \alpha_{raw} - 360 & (\alpha_{raw} \geq 180) \\ \alpha_{raw} & (\text{otherwise}) \end{cases} \quad (1)$$

下記の式 (2) によって線形補完を行うことでポイントングする画面座標を算出する。ただしディスプレイの座標は左手系 (ディスプレイ左上を原点とする) で、x, y はポイントングする画面の x 座標、y 座標とし、 α , β は現在の α 値、 β 値、 w_1 , h_1 はディスプレイの画面左上の x 座標、y 座標、 w_2 , h_2 はディスプレイの画面右下の x 座標、y 座標、 α_1 , α_2 は第 1 点、第 2 点における α 値、 β_1 , β_2 は第 1 点、第 2 点における β 値を表している。

$$\begin{aligned} x &= w_2 + \frac{w_1 - w_2}{\alpha_2 - \alpha_1}(\alpha - \alpha_1) \\ y &= h_1 + \frac{h_2 - h_1}{\beta_2 - \beta_1}(\beta - \beta_1) \end{aligned} \quad (2)$$

このようにして得られる平面なディスプレイの座標は、

*1 <https://w3c.github.io/deviceorientation/> (参照 2020-11-09)
*2 <https://java.com/ja/> (参照 2020-11-09)

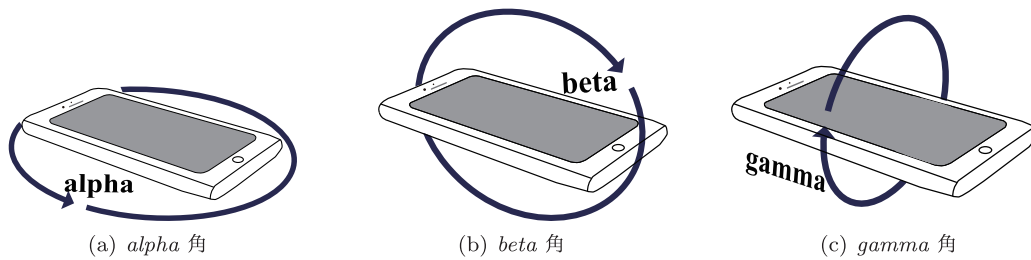


図 3 スマートフォンにおける座標系

Fig. 3 The coordinate system in a smartphone.

スマートフォンの回転角をもとに算出しているため、角度によって座標にゆがみが生じることが考えられる。このゆがみはディスプレイに近いほど、ディスプレイの中心から離れるほどに大きくなるが、一般的に我々はディスプレイを中心付近から一定の距離を保って利用することが多い。提案手法のこの構造上のゆがみは、ディスプレイの中心、ディスプレイの高さの3倍の距離から利用した場合最大でディスプレイの幅の2パーセント程度である。一方で、ユーザによるポインティング動作は約9°の誤差があることが報告されており [4]、提案手法の持つ誤差は相対的に小さくなる。そのためユーザは提案手法の持つゆがみをほとんど認識することなく利用することが可能であると考えられる。

また、複数人による接続は各ユーザのスマートフォンに固有の識別番号を割り振り、その番号ごとにポインティングする画面座標を算出することで実現した。

4.1.3 中継サーバ

スマートフォン側アプリケーションとディスプレイ側アプリケーションの通信を中継するため、Node.js 6.10.2^{*3}で構築したサーバをネットワーク上に設置する。このサーバの役割は、スマートフォンから受信したデータをディスプレイに、ディスプレイから受信したデータをスマートフォンに送信することである。ただしサーバを外部（グローバル）ネットワーク上に設置した場合、通信にかかる時間が増加し、操作の遅延につながる。そのため、スマートフォンおよびディスプレイと同一（ローカル）ネットワーク上に設置することが望ましい。

4.2 利用手順

提案手法はユーザがキャリブレーションを行うことで、ユーザがポインティングを行っている座標を計測することが可能となる。キャリブレーション時に、ユーザがとる行動とシステムが行う処理を以下に示す。

- (1) ユーザはスマートフォンを用いて、コントローラページにアクセスする。コントローラページのURLはQRコードとして提示されており、URLクエリパラメータとして中継サーバのアドレスを記述している。そのためユーザはスマートフォンを用いてQRコードを読

み込むだけでサーバへの接続を行うことができる。

- (2) ユーザはスクリーンの右上、左下へスマートフォンを向け、それぞれで画面をタップする。システムはタップされた瞬間の回転角を記録する。
- (3) ユーザはスクリーン上の任意の位置にスマートフォンを向ける。システムは回転角をもとに座標を算出する。

類似する手法として EasyPointer [7] がある。EasyPointer では基準点を4点としてキャリブレーションを行う。しかし、ユーザによるポインティング動作は約9°の誤差がある [4] ため、我々はユーザが厳密なキャリブレーションを行うことは困難であり、4点によるキャリブレーションは不要と考え、基準点を2点とした。基準点を減らしたことでユーザにかかるキャリブレーションの負担の軽減が実現できる。

5. 実験

提案手法が複数人によるポインタの同時操作を実現しているか検証する。まず、ポインタの数が増加した場合ユーザが自身のポインタを認識することができるか実験を行う。また、提案手法がポインタ操作手法としての機能を満たしているか確認する。いずれの実験でも提案手法と光学マウス（以下、マウスと記す）、ジャイロマウスを比較する。

実験を行うため、提案手法によってマウスポインタを操作できるシステムを開発した。このシステムでは算出されたポインティング座標へマウスポインタが移動する。

実験には研究室外の10代から50代の男女10名が参加し、謝金はなかった。実験は、5.2節および5.3節に示す2つの内容について実施し、実験の順番は被験者を均等に二分割しそれぞれのグループで実施する順番を入れ替えた。1人あたりの実験所要時間は30分程度で、実験の途中で中断が可能であることは被験者に伝えていたが、途中休憩などを希望した被験者がいなかったため、中断なく実験を行った。このため、疲労による実験結果への影響は発生しなかったと考えている。

5.1 実験環境

ユーザの入力動作とコンピュータへのポインティングとの間に遅延が生じた場合、操作性が低下することが

^{*3} <https://nodejs.org/ja/> (参照 2020-11-09)

MacKenzie らによって報告されている [6]。4.1.3 項で述べたように、提案手法では中継サーバを設置する場所により操作に遅延が発生すると考えられる。そこで本稿における実験ではローカルネットワーク上にサーバを設置する。これによって通信に起因する遅延を最小化し、理想環境下における提案手法の性能を測る。通信速度による遅延問題は存在しているが、ここではポインティング性能のみに焦点を当てる。

実験環境における遅延時間をラウンドトリップタイム（スマートフォン側アプリケーションからデータを送信し、ディスプレイ側アプリケーションがそれを受信後ただちに送り返し、スマートフォン側アプリケーションにてそのデータが確認されるまでの所要時間）をもとに測定した。測定は実験に用いる機材を使用して行い、ラウンドトリップタイムを 1,000 回測定し、その平均の半分の時間を通信による遅延とした。この測定の結果、遅延は平均 13 ミリ秒程度であった。

それぞれの実験でマウスポインタを操作する端末として、マウスは ELECOM 社 M-IR01DR *4 を、ジャイロマウスは Logicool 社 Spotlight Presentation Remote *5 を、提案手法は Apple 社 iPhone7 *6 を用いた。実験システムは Windows 7 OS を使用して実行し、マウスのオプションはデフォルト設定（ポインタの速度を中間値、ポインタの精度を高めるオプションを有効）として実験を行った。

5.2 複数のポインタが表示されている場合のユーザの認識精度を検証する実験

ディスプレイ上に多数のユーザが操作する複数のポインタが表示されている場合、ユーザは自身のポインタを見失い操作が困難になることがあると柴崎らの研究で報告されている [19]。また、渡邊らの研究 [13] では、同一ディスプレイ上に複数のポインタを表示した場合、ユーザは自身のポインタの認知に時間がかかることが示唆されている。しかし、これら先行研究は相対ポインティング手法における結果であり、絶対ポインティング手法である提案手法ではユーザの認知が容易になると考えられる。そこで、システムを同時に利用するユーザが増加し、ディスプレイ上に多数のポインタが表示された状況を想定し、ユーザは自身のポインタを認識することが可能か、またそれにかかる時間を実験を通して明らかにする。

5.2.1 実験手続き

実験において実際に複数の被験者が同時にポインタを操作した場合、同時に試行を行う被験者の組み合わせが実験結果に影響を与えることが想定され、これは再現性を損な

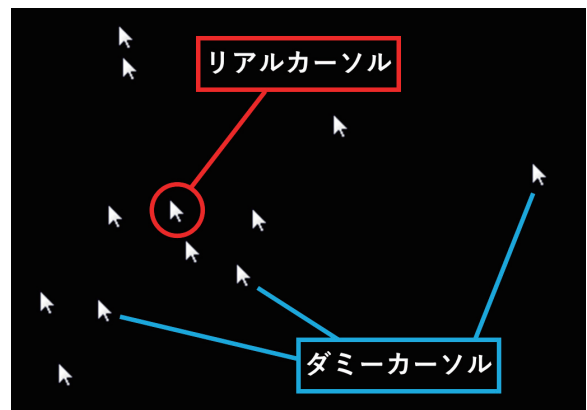


図 4 自身のカーソルを発見する実験システム

Fig. 4 The experiment system for find the real cursor.

うことにつながる。そこで、ディスプレイ上に複数のユーザが操作するポインタが存在する環境を、乱数を用いて擬似的に再現する。渡邊らがマウスを用いて行った実験 [13] を参考に実験を行う。

実験ではディスプレイ上に複数のポインタを表示する。そのうちのただ 1 つがユーザの操作によって移動するポインタ（以下、リアルカーソルと記す）であり、その他のポインタ（以下、ダミーカーソルと記す）はランダムに動く。またリアルカーソルが停止した場合、すべてのダミーカーソルも同様に停止する。渡邊らの実験でダミーカーソルは事前に登録されていたボタンを用いて実装されていたが、ここでは再現性を保つためランダムな動作を用いた。これによりダミーカーソルが不自然な動きを行う可能性があり、ダミーカーソルの数と実際に同時利用するユーザの許容数とが一致しないことが想定される。しかし手法どうしの比較を目的とする限りにおいては、同環境下で実験を行うため、評価を行うことができると考えられる。実験システムの様子を図 4 に示す。

実験参加者には複数のカーソルから、リアルカーソルを探すタスクを課す。参加者がリアルカーソルを正しく発見できたか、また発見できた場合は発見までにかかった時間を計測し、評価する。実験ではダミーカーソルの数を 10, 20, 50, 100 個とし、参加者はそれぞれの個数で 5 回の計測を行った。実験の手順を以下に示す。

- (1) 入力デバイスの使用方法を説明し、実験参加者が正しく使用できるか確認する。
- (2) 参加者がキーボードのスペースキーを入力することでシステムが稼動し、同時に時間の計測が始まる。
- (3) 参加者が入力デバイスを操作し自身のカーソルを探す。
- (4) 参加者が自身のカーソルを発見した場合、参加者はスペースキーを入力する。これによって計測が停止する。
- (5) 実験者が参加者に、自身のものだと感じたカーソルを尋ね、実験者がその正否を確認する。

実験を行うデバイスの順序は偏りがないよう被験者ごとにランダムに変更し、順序による影響を排除した。また、

*4 <https://www2.elecom.co.jp/peripheral/mouse/m-ir01dr/> (参照 2020-11-09)

*5 <https://www.logicool.co.jp/ja-jp/product/spotlight-presentation-remote> (参照 2020-11-09)

*6 <https://www.apple.com/jp/iphone/> (参照 2020-11-09)

表 1 自身のカーソルを発見する実験の正誤認識回数

Table 1 Number of correct/false recognition of experiment to find the real cursor.

	正解	誤認識	計
提案手法	194	6	200
マウス	184	16	200
ジャイロマウス	186	14	200
計	564	36	600

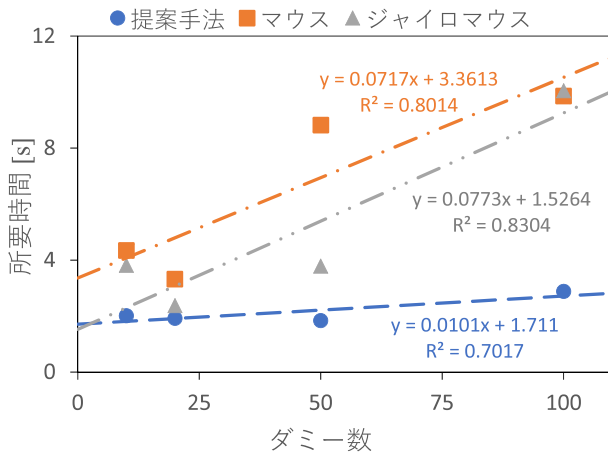


図 5 自身のカーソル発見の実験結果

Fig. 5 Results of the experiment to find the real cursor.

ダミーカーソルの数は少ない個数から順に実施し、それぞれ連続して5回の反復試行を行った。これは実験に先立ち実施した実験システムの検証において、はじめに100個のダミーカーソルが表示された場合に被験者が戸惑うことがあったためである。順序を固定することで結果に影響が出ることが想定されるが、操作手法は実験の実施順序に偏りがないため、手法間の比較を目的とした実験は可能であると考えられる。

5.2.2 結果と考察

合計600回の試行結果が得られた。カーソル認識の正誤認識回数を表1に示す。正誤認識確率について多重比較 (Bonferroni の手法) を行った結果有意差は認められなかった。

次にカーソルの認識にかかる時間について、得られた実験結果から誤認識した場合を排除したものを対象に検証を行った。ダミー数と、リアルカーソルの発見にかかった時間の関係を図5に示す。実験においてダミー数は全被験者が同一の順序で試行を行ったため、順序による影響が想定されたが、図5においてその様子は確認されなかった。これらについて二元配置反復測定分散分析を行った結果、ポインタ操作手法とダミー数に交互作用が認められた ($p = .02 < .05$)。ポインタ操作手法とダミー数において多重比較を行った結果、ダミー数50で提案手法とマウスの間に ($p = .005 < .05$)、ダミー数100で提案手法とマウス ($p = 5.5 \times 10^{-4} < .05$) および提案手法とジャイロマウス

($p = 1.7 \times 10^{-4} < .05$) の間に有意差が認められた。これらから、ダミー数が増加した場合提案手法が有意にリアルカーソルの発見にかかる時間を抑えていることが示された。また、ポインタ操作手法間で多重比較 (Bonferroni の手法) の結果、提案手法とマウス ($p = 1.0 \times 10^{-6} < .05$)、提案手法とジャイロマウス ($p = 7.9 \times 10^{-3} < .05$) の組み合わせで有意差が認められた。

このように認識にかかる時間に有意差が生じた理由は、ユーザの動作とポインタとの関連性にあると考えられる。マウスやジャイロマウスは操作量に応じてポインタが移動する相対ポインティング手法である一方、提案手法は端末で示した先にポインタが表示される絶対ポインティング手法であるため、身体動作と強く関連がありポインタの発見を容易にしたと考えられる。

5.3 ポインティング性能を評価する実験

提案手法がポインティング手法として利用可能な性能であるかを計測する。スマートフォンのジャイロセンサを用いたポインティング手法は Siddhpuria らによってすでにある程度操作可能であることが示されている [8]。しかし本研究では2点を基準点とするキャリブレーションを行うため、先行研究とは条件が異なる。

5.3.1 実験手続き

Fitts はポインティングの操作時間と操作精度に相関があることを示し、モデル化した [2]。ポインティングの性能を測る実験においては、式 (3) に示す Fitts のモデルを MacKenzie が定式化した Shannon の定式が広く利用されている [5]。

$$MT = a + b \log_2 \left(\frac{A}{W} + 1 \right) \quad (3)$$

式 (3) における MT は操作にかかった時間、 A は移動距離、 W はターゲット幅である。 a と b は実験によって決まる定数であり、 $\log_2(A/W + 1)$ はポインティングの難度 ID (Index of Difficulty) を表している。

ポインティング性能を示す指標 TP (throughput) は、 ID/MT によって求めることができる [9]。 TP の単位は bps で、値が大きいほどポインティング性能が優れている。

実験では ISO 9241-9 に基づき MacKenzie が開発した実験システム^{*7}を利用する。この実験では参加者に、円形に配置されたターゲットへマウスポインタを合わせるタスクを課す。移動距離 (A) とターゲット幅 (W) の2つのパラメータを変化させることで、ポインティング難度 (ID) が変化する。いくつかのポインティング難度 (ID) における操作時間 (MT) を測定することでポインティング性能 (TP) を算出する。実験システムと各パラメータを図6に

^{*7} <http://www.yorku.ca/mack/FittsLawSoftware/> (参照 2020-11-09)

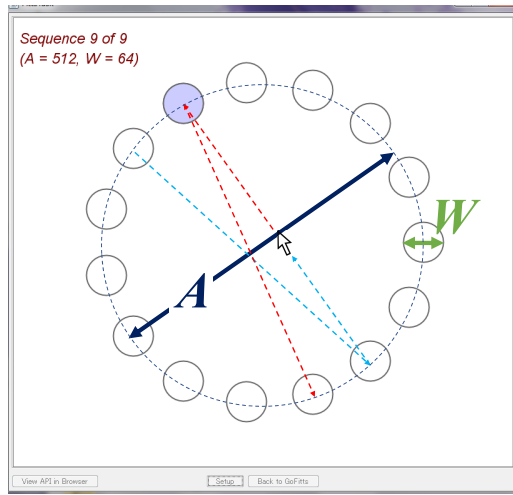


図 6 Fitts の実験システムとパラメータ

Fig. 6 Fitts' law experiment system and parameters.

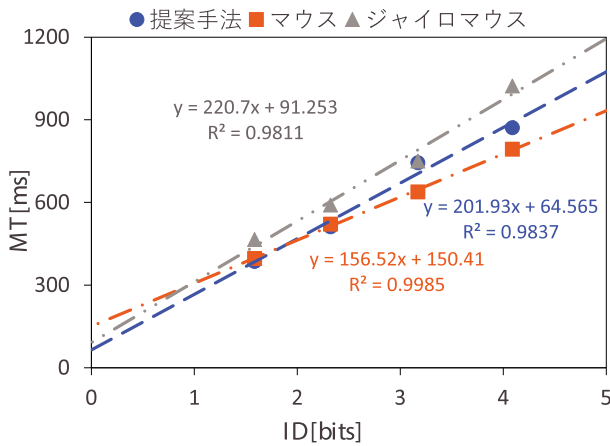


図 7 Fitts の実験結果

Fig. 7 Results of the Fitts' law experiment.

示す。

実験ではパラメータは ID が重複しないよう、移動距離 A を 128 px ないし 512 px, ターゲット幅 W を 32 px ないし 64 px とし、ターゲットの個数を 15 個とした。実験の手順を以下に示す。

- (1) 入力デバイスの使用方法を説明し、実験参加者が正しく使用できるか確認する。
- (2) 参加者は、実験システムに配置された開始ボタンを選択し、タスクを開始する。
- (3) 参加者は、実験システムに表示される円形のターゲット上へ、素早くポインタを移動する。
- (4) 正しくターゲット上にポインタが移動した場合、ターゲットは次のものへと移る。
- (5) 参加者は移動したターゲット上へポインタを移動する操作を繰り返す。

5.3.2 結果と考察

合計 1,800 回の試行結果が得られた。ポインタング難度 ID と、被験者の平均操作時間 MT の関係を図 7 に示す。

実験結果のうち TP について反復測定分散分析を行った結果、ポインタ操作手法に主効果が認められた ($F_{2,78} = 10.13$, $p = 1.2 \times 10^{-4} < .05$)。多重比較 (Bonferroni の手法) の結果、提案手法とジャイロマウス ($p = 4.6 \times 10^{-4} < .05$)、マウスとジャイロマウス ($p = 5.2 \times 10^{-4} < .05$) の組み合わせで有意差が認められた。

ポインタング性能を示す指標 (TP) は、提案手法が $TP = 4.20$, マウスが $TP = 4.45$, ジャイロマウスが $TP = 3.71$ となり、マウスのポインタング性能が最も高い結果となった。

マウスと提案手法が似た傾向 (近似曲線) を示しており、提案手法が普及しているマウスと同程度の操作性を持つと考えられる。またマウスが最も優れた結果を示したことについて、マウスは一般的なポインタングデバイスとして本実験における比較対象としたが、実験参加者の多くがマウスのみ操作経験があったことや、マウスが固定平面上で操作する端末であり、空中で操作する提案手法やジャイロマウスに比べ安定した操作が可能であったためこのような結果になったと考える。本実験では提案手法が同様に空中で操作するジャイロマウスに比べ優位に優れた結果であったことから、提案手法のポインタングシステムとしての性能は担保されていると考える。今後、携帯端末をスワイプすることによるポインタ操作手法など、空中で操作できる他の手法とさらなる比較を行い、提案手法の性能をより詳細に検証する必要がある。

6. おわりに

本稿では複数のユーザがスマートフォンによって ad hoc にポインタを操作できる手法、BYOP を提案した。提案手法では普及しているスマートフォンに搭載されているセンサのみを活用することでユーザが特殊な機器を準備する負担を軽減し、絶対ポインタング手法を実現することでユーザが増加し画面上に表示されるポインタの数が増加した場合でも自身のポインタを見失うことなく操作できるシステムを開発した。実験では、ダミーカーソル中からリアルカーソルを見つける実験から自身のポインタを発見できることを、ISO 9241-9 に基づいた実験からポインタングシステムとしての性能を検証し、それぞれ一定の有効性が確認できた。

今後の課題として、提案手法のさらなる詳細な検証の必要性が挙げられる。本稿の実験ではポインタの操作性と、ポインタの認識のそれぞれについて個別に検証を行ったが、実際の利用時には多数のユーザが操作するポインタの中でポインタを操作することが想定される。このため、複数のポインタ中における操作性を検証することが必要である。また、提案手法にはキャリブレーション後にユーザが移動した場合にスマートフォンとディスプレイとの位置関係がずれ、ポインタング座標に誤差が生じる問題があ

る。この問題は加速度センサによってユーザの移動を計算することで緩和できると考えているが、問題の程度やユーザの移動認識の精度を検証し、解決方法を探る必要がある。

参考文献

- [1] Babic, T., Reiterer, H. and Haller, M.: Pocket6: A 6DoF Controller Based On A Simple Smartphone Application, *Proc. 6th Symposium on Spatial User Interaction*, pp.2–10 (2018).
- [2] Fitts, P.M.: The information capacity of the human motor system in controlling the amplitude of movement, *Journal of Experimental Psychology*, Vol.47, No.6, pp.381–391 (1954).
- [3] Julian, S., Andreas, B. and Enrico, R.: PointerPhone: Using mobile phones for direct pointing interactions with remote displays, *Human-Computer Interaction – INTERACT 2013*, Vol.8119 LNCS, No.PART 3, pp.18–35 (online), DOI: 10.1007/978-3-642-40477-1_2 (2013).
- [4] Kang, R., Guo, A., Laput, G., Li, Y. and Chen, X.A.: Minuet: Multimodal Interaction with an Internet of Things, *Symposium on Spatial User Interaction*, pp.2:1–2:10 (2019).
- [5] MacKenzie, I.S.: A Note on the Information-Theoretic Basis for Fitts' Law, *Journal of Motor Behavior*, Vol.21, No.3, pp.323–330 (1989).
- [6] MacKenzie, I.S. and Ware, C.: Lag As a Determinant of Human Performance in Interactive Systems, *Proc. INTERACT '93 and CHI '93 Conference on Human Factors in Computing Systems*, pp.488–493 (1993).
- [7] Pan, G., Ren, H., Hua, W., Zheng, Q. and Li, S.: Easy-Pointer: What You Pointing at is What You Get, *CHI '11 Extended Abstracts on Human Factors in Computing Systems*, pp.499–502 (2011).
- [8] Siddhpuria, S., Malacria, S., Nancel, M. and Lank, E.: Pointing at a Distance with Everyday Smart Devices, *Proc. 2018 CHI Conference on Human Factors in Computing Systems*, No.173, pp.1–11 (2018).
- [9] Soukoreff, R.W. and MacKenzie, I.S.: Towards a standard for pointing device evaluation, perspectives on 27 years of Fitts' law research in HCI, *International Journal of Human Computer Studies*, Vol.61, No.6, pp.751–789 (2004).
- [10] Sweetser, J., Grunnet-Jepsen, A. and Panchanathan, G.: Absolute Pointing and Tracking Based Remote Control for Interactive User Experience, *Proc. 1st International Conference on Designing Interactive User Experiences for TV and Video*, pp.155–164 (2008).
- [11] Wilson, A. and Shafer, S.: XWand: UI for Intelligent Spaces, *Proc. SIGCHI Conference on Human Factors in Computing Systems*, pp.545–552 (2003).
- [12] 牛田啓太, 村田雄一: ペンライトを用いた 10 フィート/大画面向けユーザインタフェース, *映像情報メディア学会誌*, Vol.62, No.1, pp.126–131 (2008).
- [13] 渡邊恵太, 樋口文人, 稲見昌彦, 五十嵐健夫: 複数ゲームカーソル中における自分自身のカーソル特定, *情報処理学会インタラクション 2013 論文集*, pp.25–31 (2013).
- [14] 天早健太, 木戸瑛一, 杉原慶哉, 中道 上, 渡辺恵太, 山田俊哉: Remote Touch Panel: 大画面における直観的なタップジェスチャー, *インタラクション 2018 論文集*, pp.779–784 (2018).
- [15] 新谷晃市, 間下以大, 清川 清, 竹村治雄: 大画面ポインティングシステムのための回帰モデルによる単眼画像からの指差し位置の推定, *情報処理学会研究報告*,

Vol.2009-CVIM-167, No.33, pp.1–8 (2009).

- [16] 牟田将史, 益子宗, 新里圭司, Mujibiyah, A.: Wall-SHOP: Web ブラウザのみで動作する複数人が携帯端末から操作可能なデジタルサイネージ, *インタラクション 2015 論文集*, pp.826–831 (2015).
- [17] 宮田章裕, 瀬古俊一, 青木良輔, 橋本 遼, 石田達郎, 伊勢崎隆司, 渡辺昌洋, 井原雅行: デジタルサイネージとモバイル端末を連携させた複数人同時閲覧のための情報提示システム, *情報処理学会論文誌*, Vol.56, No.1, pp.106–117 (2015).
- [18] 瀧口智史, 田中敏光, 佐川雄二: Web カメラを用いた片手文字・座標入力システム, *情報処理学会研究報告*, Vol.2010-UBI-2, No.11, pp.1–6 (2010).
- [19] 柴崎美奈, 張 衍義, 武田 港, 小幡光一, 吉野弘一, 遠藤志津子, 徳久 悟, 南澤孝太: VR シアターにおけるパーソナルな鑑賞体験を支援する多人数ポインティング, *日本バーチャルリアリティ学会論文誌*, Vol.20, No.4, pp.333–344 (2015).



佐藤 光起

2019 年関西大学総合情報学部総合情報学科卒業。同年関西大学大学院総合情報学研究科知識情報学専攻博士前期過程入学, 現在に至る。ヒューマンコンピュータインタラクションの研究に従事。



松下 光範 (正会員)

1995 年大阪大学大学院基礎工学研究科物理系専攻制御工学分野博士前期課程修了。同年日本電信電話株式会社入社。2008 年関西大学総合情報学部准教授。2010 年同教授。自然言語理解, インタラクションデザインに関する研究に従事。博士(工学)。2003 年情報処理学会論文賞, 2013 年 LavalVirtual Award ほか各賞受賞。電子情報通信学会, 人工知能学会, 芸術科学会, ACM 各会員。