

# タッチ操作ログに基づいた操作形態推定手法

平部 裕子<sup>1</sup> 諏訪 博彦<sup>1,2,a)</sup> 荒川 豊<sup>3</sup> 安本 慶一<sup>1</sup>

受付日 2021年4月12日, 採録日 2021年10月8日

**概要:** ユーザがどのようにスマートフォンを持ち、どの指で操作しているかという情報（操作形態）が分かれば、様々な新しい応用、たとえば、操作形態に適したUI（ユーザインターフェース）への動的切替えや、ながらコンテキスト（スマートフォン操作時に行われている別行動、歩行、喫煙、食事など）の推定など、への道が拓ける可能性がある。本論文では、タッチ操作によって出力されるログに着目し、タッチやスワイプなどの操作領域や頻度からどのような操作形態かを推定する方法を提案する。そのために、Android上でアプリ横断的にタッチ操作を取得可能な TouchAnalyzer から得られた情報を特徴量とし、機械学習手法により代表的な8通りの操作形態を推定する手法を構築する。16名の実験参加者による評価実験を通じて、提案手法によって操作形態をおおむね90%のF値で推定できることが分かった。

キーワード: タッチ操作ログ, スマートフォン, 操作形態推定, 機械学習

## Estimating Phone Holding Style through Analysis of Touch Logs

YUKO HIRABE<sup>1</sup> HIROHIKO SUWA<sup>1,2,a)</sup> YUTAKA ARAKAWA<sup>3</sup> KEIICHI YASUMOTO<sup>1</sup>

Received: April 12, 2021, Accepted: October 8, 2021

**Abstract:** In this paper, we propose a method for estimating the *phone holding style* that shows hand grasp and finger placement while using a smartphone (e.g., either by both hands or one hand or by putting it on the desk) through analysis of screen touch behavior logs. Knowing the phone holding style in real-time will open the door to various new applications such as dynamically providing a suitable interface (e.g., layouts of buttons) and estimating “side-contexts” like walking, smoking and drinking while using a smartphone. In this paper, focusing on the logs that Android operating systems generate when touch behavior happens, we propose a method to estimate the way of holding a smartphone by area (in the screen) and frequency of touch behavior. We have developed a tool called *TouchAnalyzer* that identifies touch behavior types like swipe, rotate, pinch, multi-touch, etc. from the device-dependent logs of Android OS. By using the touch behavior type logs that TouchAnalyzer outputs, we have developed an algorithm that can recognize eight representative phone holding styles. We conducted evaluation experiments with 16 participants and confirmed that the proposed method achieves about 90% of F-score in estimating the phone holding style.

**Keywords:** touch behavior logs, smartphone, phone holding style estimation, machine learning

### 1. はじめに

持続可能な社会を実現するためには、多様なユーザの状態（コンテキスト）や属性（プロファイル）を推定しな

で、各ユーザにあった支援を実施するコラボレーション技術が重要となる。多様化するユーザに対して、適切な支援・機能（サービス、広告、UI (User Interface) など）を提供するには、ユーザのコンテキストやプロファイルを何らかの方法で認識する必要がある。ユーザのコンテキストやプロファイルを推定するためのインプット情報は様々考えられるが、我々は多くの人が利用するスマートフォンに焦点を当てる。

コンテキストに関しては、スマートフォンに内蔵されたモーションセンサを用いることで動作を検出し歩行時や着

<sup>1</sup> 奈良先端科学技術大学院大学

NAIST, Ikoma, Nara 630-0192, Japan

<sup>2</sup> 理化学研究所革新知能統合研究センター AIP

RIKEN, Center for Advanced Intelligence Project AIP, Chuo, Tokyo 103-0027, Japan

<sup>3</sup> 九州大学

Kyushu University, Fukuoka 819-0395, Japan

a) h-suwa@is.naist.jp

座時に応じた機能提供する<sup>\*1,\*2</sup> [1] といった試みがなされている。また、位置情報を用いて、位置に応じたサービスを提供するといったサービス<sup>\*3,\*4</sup> [2], [3] も多数提案されている。

一方、年齢層や性別、利き手、操作スキルなどのユーザのプロファイルに関して、スマートフォン上で認識を試みた研究や実サービスは様々存在する。たとえば Google は、検索履歴に基づくユーザプロファイリングを実施<sup>\*5</sup>しており、年齢や性別、興味などを推定し、適切な広告を提示する仕組みを構築している。しかし、この対象はあくまで Google 検索利用者限定される。また、通信キャリアにおいては、契約情報からこれらの属性情報の一部を取得することは可能であるが、個人情報保護の観点から、スマートフォンの端末メーカーにその情報を提供することは難しい。さらにアプリのデザイナーや開発者においては、適切なデザインの設計や比較と検証、およびアプリバグの原因追求が重要な課題である。この課題に対して、ユーザの利き手や、操作スキル、操作形態（どのようにスマートフォンを持ち、どの指で操作しているかという情報）といったユーザプロファイルは有用な情報であり、実際にアイコンやボタン、スワイプの利用情報（アプリ操作情報）を取得し改善につなげている開発者も多い。しかしながら、アプリ操作情報の取得にも、アプリ独自にスクリプトを組み込む必要があり、利用者の実データを用いたアプリ横断のデザイン設計に関する比較や検証、バグ原因追求は困難である。いずれの場合においても、ユーザプロファイルの中でもベーシックな、実際にユーザがスマートフォンをどのように操作しているのかを示す操作形態を取得することはできない。

操作形態を認識する研究として、Goel らは GripSense [4] を提案している。しかしながら、Gripsense は独自のアプリケーション上で動いているため、単一アプリ内ではしかデータ取得ができない。そのためアプリケーションによらず端末利用者の操作情報を把握し、統合的に操作形態を識別することは難しい。他の手法としては、静電タッチセンサ [5], [6] や Grip センサを用いて認識する手法 [7], [8], 端末カバーに光センサを多数組み込むことで操作形態を認識している研究 [9] も存在するが、いずれも追加のセンサが必要となり、スマートフォン単体で利用できない。また、文字入力に着目した TouchML [10] や ContextType [11] などの研究もあるが、タップ操作による文字入力に注力しており、スワイプやマルチタッチといったスマートフォンならではの、かつ、アプリケーションによって変化する操作は考慮されていない。

本論文では、アプリ横断的かつ端末非依存である（要件

1), ユーザや端末の負荷が少ない (要件 2), の 2 つの要件を満たす操作形態推定手法を提案する。提案手法は、タッチ操作によって出力されるログに着目し、タッチやスワイプなどの操作領域や頻度からどのような操作形態かを推定する。我々は、Android 上でアプリ横断的にタッチ操作を取得可能な TouchAnalyzer を開発している [15], [18]。本研究では、開発した TouchAnalyzer から得られた情報を特徴量として、機械学習手法により操作形態を推定するための手法を構築する。機械学習アルゴリズムには、決定木、ロジスティック回帰、ランダムフォレストの 3 種類を使用し、それぞれの推定精度を評価する。

TouchAnalyzer を用いてスマートフォンの使い方を観察したところ、経験的に操作形態によってタッチ開始/終了位置、タッチ頻度やスワイプ軌跡の曲がり具合が異なることが分かった [16], [17]。また、関連研究でも指摘されているとおり [12], 手のサイズが操作に大きな影響を与えることが分かった。そこで、タッチ開始/終了位置の x 軸および y 軸、X 軸のタッチ頻度、Y 軸のタッチ頻度などのタッチ情報と手のサイズを特徴量として使用することとした。16 名の実験参加者から得られたログに基づいて、操作形態推定を行った結果、ランダムフォレストを用いた際におおむね 90% の F 値で各操作形態を識別可能であることが分かった。

本研究の貢献は、タッチ操作ログから操作形態を推定できるかというリサーチクエスチョンに対し、どんな操作形態を推定すべきか、推定に有効な特徴量は何か、推定に有効な機械学習アルゴリズムは何か、スマートフォンの大きさによって影響があるか、というサブクエスチョンを立て、それぞれ明らかにしていることである。具体的には、推定する操作形態の定義を整理したこと、指の長さが操作形態の推定に有効な特徴量であること、ランダムフォレストが操作形態の推定に有効であること、スマートフォンのサイズによって推定精度が変化すること、を明らかにしていることである。

## 2. 関連研究

スマートフォンの操作形態の認識に関して、これまで様々な研究が行われている。本章では、これら関連研究の概要を述べ、本研究との差異を明らかにする。

### 2.1 組み込みセンサや追加センサを利用した推定手法

スマートフォンの組み込みセンサを利用して、ヒューリスティックに操作形態を推定する手法に、Goel らの GripSense [4] がある。GripSense はスマートフォンの加速度センサやジャイロセンサ、振動センサを利用して、タッチパネル入力時に起こる端末の傾きや、スワイプ、タッチサイズ、振動を各被験者ごとに計測して、端末を握る右左の手や親指と人差し指、圧力、端末が机の上に置かれてい

\*1 [https://www.nttdocomo.co.jp/service/anshin\\_mode/](https://www.nttdocomo.co.jp/service/anshin_mode/)

\*2 <https://www.au.com/mobile/service/aruki-sumaho/>

\*3 <https://k-tai.sharp.co.jp/dash/emopa/introduction/base/>

\*4 <https://www.cyberagent.co.jp/news/detail/id=11495>

\*5 <https://www.google.com/settings/ads/onweb/?hl=ja>

る, といった情報を含む操作形態をヒューリスティックに推定している. Lochtefeldら [13] も画面ロック解除時に GripSense と同様の情報を取得および利用して, 操作形態の推定を行っている. しかしこれらの手法は, 専用のアプリケーション上でしか利用できないため, アプリケーションによらず端末利用者の操作情報を取得して, 統合的に操作形態を識別することはできない. また QWERTY キーボードやロック解除画面からの入力を利用しているため, 日本語入力によく使われるフリックキーボードや UI が異なるアプリケーションには適用できない.

スマートフォンに独自のセンサを組み込むことで, 操作形態を認識する研究も多数存在する. たとえば, Harrisonら [5] や Kimら [6] は, 静電タッチセンサを対象デバイスに組み込むことで操作形態の認識を可能にしている. また端末に Grip センサを組み込む手法 [7] も存在する. さらに, これらのセンサと加速度センサを組み合わせることで, 操作形態を認識する研究 [8] も存在する. これらの手法は, 短い期間に持ち方を認識できる有効な手法であるものの, スマートフォンに独自のセンサを組み込む必要があり, 汎用的ではない.

## 2.2 機械学習を用いた手法

特殊なセンサを用いずに操作形態の認識を試みる研究も存在する. 多くは機械学習を用いた手法である. Buschekら [10], [14] は, 被験者 24 人から 2 週間にわたりタッチデータの収集を行い, タッチエラーの統計的パターン情報をもとに機械学習によってオフセットモデルを構築し, 操作形態の認識へと発展させている. また GripSense [4] では, スマートフォンの加速度センサやジャイロセンサ, 振動センサを利用して, タッチパネル入力時に起こる端末の傾きやタッチサイズ, 振動を各被験者ごとに計測し, その情報を機械学習によってモデル化し, 操作形態の情報を推定している. さらに, ContextType [11] は, GripSense の仕組みを利用して推定した操作形態情報とタッチキーボードへの入力情報と組み合わせて, 操作している手を通知し入力ミスの改善を提案している.

しかしながら, 機械学習を利用したこれらの手法では, 長期間のデータ収集を必要としたり, 被験者ごとに学習データを必要とするため, ユーザに与える負荷が大きい.

## 2.3 本研究の狙い

従来手法の問題点として, 特殊な追加センサを使用するため汎用性が保てない点, ユーザの負荷が大きい点があげられる. 本研究では, 追加のセンサを使用せず, アプリケーションおよび端末非依存で, かつユーザの負荷が少ないスマートフォンの操作形態認識手法を提案するという点で既存研究と異なっている.

## 3. タッチ操作ログ分析に基づく操作形態推定手法の提案

### 3.1 操作形態の定義

我々は, スマートフォンの操作形態を把握することにより, アプリのデザインや操作形態に合わせた UI (ユーザインタフェース) の動的切替えや, ながらコンテキストの推定精度向上に貢献できると考えている. たとえば, UI の動的切替えであれば, 画面の左上が押しにくい操作形態 (右手持ち右手親指操作) のときに, よく使うアプリやアイコンが押しやすい場所に移動してくるような UI を実現できると考える. また, 操作形態はユーザのコンテキスト変化にともない変化する. たとえば, 着席時や仰臥位 (仰向け) のときには両手操作 (一方の手で持ちもう一方の手で操作), 歩行時や電車内で起立時は片手操作, 食事中などは卓上にスマホを置いて操作するなどが考えられる. そのため, 操作形態の認識はながらコンテキスト (スマートフォン操作時に行われている別行動, 歩行, 喫煙, 食事など) の認識に役立つと考える.

スマートフォンの操作形態は, 人により様々であり, 右手で支え, 右手の人差し指で操作する人も数は少ないが存在する. しかし, 数少ない操作形態を推定することは, 代表的な操作形態を推定することに比べ相対的に重要度が低いと考える. 本研究では, 代表的な操作形態に限定し, 推定を試みる. このような考え方に基づき, 推定する操作形態を定義する. 本研究では, スマートフォンの操作形態を, スマートフォンを支える手 (支える手), およびスマートフォンの画面を操作する手と指 (操作する手, 操作する指) の 3 要素の組合せ (以下, 操作形態と記述) とし, 3 要素がとりうる状態を以下のように定義する.

- 支える手 {右, 左, 無し}
- 操作する手 {右, 左}
- 操作する指 {親指, 人差し指}

ここで, 支える手や操作する手について, 右手, 左手以外に両手というケースも考えられる. 操作する指についても, 中指, 薬指, 小指も考えられる. 我々は, 代表的なスマートフォンの操作形態を確認することを目的に, 既存のスマートフォンの持ち方調査 [19], [20] を参考に主な操作形態に関してクラウドソーシングを用いた 300 人に対するアンケート調査を実施した. アンケートは, 「あなたは, スマートフォンを主にどのように操作しますか?」という質問に対して, 支える手および操作する手については, 右手, 左手, 両手のいずれかで回答し, 操作する指については, 親指, 人差し指, 中指, 薬指, 小指のいずれかで回答してもらった. 回答の信頼性を確保するためにチェック質問を設け, チェック質問に正しく回答した回答者のデータだけを収集している.

その結果, 両手で支えたり, 両手で操作したりする割合

は、それぞれ 4.3%, 2.7%と少数であったため、今回は推定する操作形態から除外することとした。また、食事中などスマートフォンが机の上などに置かれている状況を想定し、支える手がない場合を設定している。操作する指については、90%以上のユーザが親指と人差し指と回答したため、今回はユーザの指の使用頻度を考慮して親指と人差し指とした。また、マルチタッチについては、シングルタッチやスワイプと比べて使用頻度が少ないと考え、本論文では対象外としている。

定義に基づいて組合せを考えると 12 の操作形態が考えられるが、右手で支え右手の人差し指で操作する形態、左手で支え右手の親指で操作する形態、左手で支え左手の人差し指で操作する形態、右手で支え左手の親指で操作する形態は、それぞれ 2.0%, 3.7%, 1.3%, 1.0%と少数であったため除外している。結果として、端末操作者は図 1 に示すとおり、8 種類の操作形態のうちいずれかでスマートフォンを操作することとなる。図 1 では例として、支持手がある場合に起こる 4 つの操作形態に関し、イラスト化している。残り 4 つはイラスト化した操作形態が机に置かれた状態で起こる形態を指す。本研究ではこの 8 形態を推定対象とする。

### 3.2 要件定義

本研究では、追加のセンサなしに、アプリケーションを横断して端末非依存にスマートフォンのタッチ操作ログを取得、利用して、短いデータ収集期間で 8 つの操作形態を認識することを目的とし、タッチ操作ログ分析に基づく操作形態認識手法を提案する。タッチ操作は単一のアプリケーション内であれば、特殊なスクリプトを埋め込む事で簡単に取得できる。しかし Android に代表される OS (Operation System) では、セキュリティの観点から、アプリケーションを横断して情報を取得することは一般的に難しい。我々はアプリケーションによらず情報を取得して統合的に操作形態を認識することを目的としているため、この手法が利用できない。よって、本手法の要件として以下の 2 つを設定する。

#### 要件 1 アプリ横断的かつ端末非依存な操作形態の認識

アプリ横断的かつ端末非依存な操作形態の認識をするためには、タッチ操作情報をロギングする特殊なスクリプトを全アプリに埋め込む方法が考えられる。しかしこの手法は現実的とはいえない。一方でタッチ操作情報をロギングする専用アプリケーションを作成し、絶えず他のアプリを監視するといった方法も考えられる。しかし Android に代表される OS は、サンドボックスと呼ばれるセキュリティの観点から各アプリが強制的に分離、独立しているため、それぞれのアプリケーションが所有するデータを相互に見ることは、root と呼ばれる管理者権限を取得しない限り、不可能である。

支持手	操作		Class
	手	指	
右	右	親	1 →
		人差し指	—
親		—	
人差し指		2 →	
左	右	親	5
人差し指		6	
無し (卓上)	左	親	3 →
左		人差し指	—
		親	—
右		人差し指	4 →
無し (卓上)	左	親	7
		人差し指	8

図 1 操作形態の種類

Fig. 1 Types of operation forms.

これらのことから、アプリケーションを横断してタッチ操作情報を取得するには、スマートフォンの仕組み上、OS が出力する情報を取得するしかない。しかし OS が出力するデバイスログは、低レベルの文字の羅列であるうえ、端末ごとにフォーマットも異なる。たとえばスワイプ動作を点の塊として単にログを出力しているため、取得したデバイスログから、それらのログがシングル (またはマルチ) タッチなのかシングル (マルチ) スワイプか、といったタッチ操作の挙動内容や、挙動頻度や速度、操作範囲などの高レベルの情報に変換する必要がある。

要件 1 に対応するために、データ収集システム Touch-Analyzer を開発している [15], [18] (詳細は、3.3 節で述べる)。

#### 要件 2 ユーザや端末の負荷が少ないシンプルな認識手法

操作形態を認識するにあたり、ユーザや端末に負荷をかけることは望ましくない。たとえば、学習データの作成のために端末利用者が意識的にシステム利用し長期間のデータを集積することや、各ユーザによる事前の正解データ登録作業が必要となれば、ユーザに負荷がかかり満足度は低下する。

また、操作形態を認識するために、多くのセンサを利用しデータ収集する場合、データ処理量・消費電力の観点から端末の負荷が大きくなり、ユーザの利用可能時間が減少する。結果として、ユーザの満足度は低下する。同様に、複雑な計算をさせることはより多くの負荷を端末にかけることになるため、シンプルなルールで操作形態を認識することが求められる。

要件 2 に対応するために、推定モデル構築に用いる特徴量は、ユーザがスマートフォンを本来の目的のために操作した際に収集可能なデータに限定する。また、特徴量の数

を極力減らし、推定モデルを構築するためのアルゴリズムもシンプルなものを検討する。収集したデータをインターネット経由でサーバに送信することなくスマートフォン端末のみで推定できると考える。推定プロセスについては、3.4節で述べる。

### 3.3 TouchAnalyzer を用いたデータ収集

要件1に対応するため、OSが出力するタッチイベントのデバイスログを高レベルの情報（シングル・マルチタッチ、スワイプなど）に変換し、タッチ操作の挙動情報として識別するシステム TouchAnalyzer を開発している [15]。TouchAnalyzer は PC 上で動作するシステムであり、対象端末とシステムを USB 経由で接続する構成となっている。開発システムは Android SDK platform-tools に含まれる、Android Debug Bridge（以下 adb と記載）と python を用いて実現されている。

TouchAnalyzer は、`/dev/input` などのデバイスログに出力されるタッチイベントログを、adb 経由で絶えず監視して常時取得すると同時に、foreground（ユーザが操作している）アプリケーションログを取得する。

本システムは、機種ごとに異なるタッチイベントの出力先や時刻フォーマットを自動的に認識し、タッチイベントログをリアルタイムに収集・解析し、有意な情報として抽出する。これにより、ユーザのタッチ操作（シングル・マルチタッチ、シングル・マルチスワイプ、ピンチ、ローテート）のログを収集することが可能となる。

また、PC と対象端末を USB 接続することなく、ログデータをサーバへ収集可能な TouchAnalyzer 改良版を開発している [18]。TouchAnalyzer 改良版は、(1) データ収集部（Android アプリケーション）、(2) データ保存部（サーバ）、(3) データ解析部（TouchAnalyzer）で構成され、(1) から収集した生データを (2) に送信し、それらのデータを (3) で解析・保存する仕組みである。これにより USB 不要なデータ収集を実現している。対象端末にあらかじめ Android アプリケーションをダウンロード・インストールし、起動する必要があるが、起動後にユーザが操作の記録や解析をする必要はない。また、本システムは、従来手法で用いられていた複数センサを必要とせず、端末上で複雑なデータ計算処理も行わない。本研究では、この TouchAnalyzer から得られたタッチ操作ログを用いて操作形態推定手法を構築する。なお、スマホ OS 市場では Android, iOS の 2 つが大きなシェアを占めているが、本研究では研究・実験の容易性を考慮して Android 端末のみを対象に研究を行う。

### 3.4 推定プロセス

推定プロセスの概要を以下に示す（図 2）。今回、推定したい操作形態は 8 形態である。この 8 形態は、手で持つ場合と、卓上に置く場合とに大別できる。手で持つ場合は、

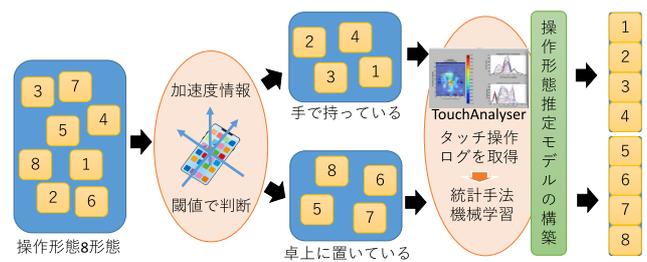


図 2 推定プロセス

Fig. 2 Process of estimation.

多少なりともスマートフォンが動くことになる。一方、卓上に置く場合は、スマートフォンはほとんど動くことなく静止している。この 2 つを分類するには、スマートフォンに内蔵されている加速度を計測すれば容易に判定できる。そこで、まず加速度を用いて 8 形態を 4 形態ずつに分ける。なお、加速度で分類する部分については、きわめて単純な仕組みのため、本論文では取り扱わない。次に、それぞれの 4 形態を TouchAnalyzer から得られるタッチ操作ログなどに基づいて推定する。推定モデルの構築には、統計手法および機械学習を用いる。なお、要件 2 に対応するためには、使用する特徴量の種類は少ないほうが望ましい。また、推定モデル構築のための数理モデルもシンプルなのが望ましい。具体的な特徴量および数理モデルについては、4 章で述べる。

## 4. 操作形態推定手法の実現

本章では、タッチ操作ログ分析に基づく操作形態推定手法を実現するための手順として、実験環境、特徴量、および評価に用いる機械学習アルゴリズムに関して詳述する。

### 4.1 実験の環境と内容

タッチ操作ログを収集するため、実験参加 16 名（男性 9 名、女性 7 名）に協力を依頼した。被験者の年齢は 23 歳から 28 歳である。

端末の大きさを考慮するため、実験端末には Nexus 5 および Nexus 6 を各 5 台ずつ利用した。実験端末の仕様情報<sup>\*6,\*7</sup>を表 1 に示す。

それぞれの実験端末にはデータ収集アプリケーションが事前にインストールされており、アプリを起動した状態で操作することで被験者のログデータを収集した。

実験参加者は、図 1 に示した 8 つの持ち方に対し 3 分間ずつ計 24 分間データを収集し、事前説明や同意書の取得、事後アンケートなどを含め実験全体で計 1 時間 30 分の実験に参加した。実験中に操作するアプリケーションは、Yahoo!ニュース<sup>\*8</sup>を指定した。Yahoo!ニュースは利用者も多く、通常よく使用される一般的なサイトと考え採用して

<sup>\*6</sup> <https://www.ymobile.jp/lineup/nexus5/>

<sup>\*7</sup> <https://www.ymobile.jp/lineup/nexus6/>

<sup>\*8</sup> <https://news.yahoo.co.jp/promo/app/yjnews/>

表 1 実験端末の仕様情報

Table 1 Status of experimental devices.

基本スペック	Nexus 5	Nexus 6
サイズ	約 W69 × H138 × D8.6 mm	約 W83 × H159 × D10 mm
重さ	約 130 g	約 184 g
画面サイズ	5.0 インチ	6.0 インチ
表示方式	TFT	有機 EL
解像度	1,080 × 1,920 ドット (フル HD)	1,440 × 2,560 ドット (クアッド HD)

いる。各参加者は、3分間アプリケーションを操作する際に操作形態を固定する以外、どのような指示も受けておらず、アプリケーションを自由に操作する。また実験前に、参加者の手の大きさを計測した。

収集ログデータのうち、評価データに、Nexus 5 に関して 10 名、Nexus 6 に対して 10 名のデータを利用した。その理由は、被験者の操作ミスによりデータ収集アプリケーションが起動しておらず、データ収集ができていなかったユーザがいたためであり、これらの被験者データは評価対象から除外している。

#### 4.2 特徴量

本研究では操作形態推定モデルを構築するために、3章の要件 2 で示した、ユーザや端末の負荷が少ないシンプルな認識手法を実現可能な特徴量を利用する。推定モデル構築に利用する特徴量はユーザのスマートフォン操作から得られるものとし、スワイプ操作の情報を利用した。スワイプ操作を選択した理由は、タッチ操作と比較してより多くの特徴量が得られると想定したためである。たとえばタッチ操作では、操作軌跡の移動量やスワイプが描く弧の方向など、操作形態に関連しうる特徴量を利用できない。評価で利用する具体的な特徴量を、以下に列挙する。

- 始点 {x 座標, y 座標}
- 終点 {x 座標, y 座標}
- 始点から終点への {x 座標の移動量, y 座標の移動量}
- スワイプが描く弧の方向
- 出現最頻領域の {x 軸方向の分割番号, y 軸方向の分割番号}

ここで、出現最頻領域について述べる。操作形態推定に利用可能な特徴量を考慮したとき、操作形態に依存して、スマートフォンのタッチパネル上で最もよく触られる領域 (出現最頻領域と定義) が出現すると考えられる。この領域を抽出するため、スマートフォンの画面領域を経験的に x 軸方向および y 軸方向へ、それぞれ 12 分割, 16 分割し、各スワイプ操作ごとに x 方向, y 方向への分割番号を導出した。

列挙した 9 つの特徴量をスワイプが行われるたびに抜き出し、すべての参加者情報をマージして、データセット 1 (今後、データ 1 と記述) を作成した。

表 2 特徴量の統計量

Table 2 Basic statistics of features.

	平均	標準偏差	最大値	最小値
始点 x	577.0	390.3	1,439	0
始点 y	1,058.2	418.0	2,559	1
終点 x	579.9	394.1	1,439	0
終点 y	1041.0	418.4	2,559	1
移動量 x	213.9	226.3	5,035	0
移動量 y	653.9	473.4	8,360	0
弧の方向	1.7	0.7	3	1
出現最頻領域 x	5.0	3.4	11	0
出現最頻領域 y	7.3	2.7	15	0
指の長さ (親指)	5.9	0.3	6.5	5.5
指の長さ (中指)	7.7	0.3	8.5	7.3
取得回数	143.0	57.9	284	29

一方で我々の先行研究 [16] で、操作している手の指の長さが操作形態の判定に利用できる可能性が有ることが分かっていた。そのためデータ 1 に、操作している手の指の長さ (親指, 中指) を特徴量として追加し、データセット 2 (以降、データ 2 と記述) を作成した。表 2 に特徴量の統計量を示す。各特徴量の、平均、標準偏差、最大値、最小値を示している。また、取得回数は、3分間で収集されたデータのサンプル数である。各参加者のデータ数をそろえることはしていない。

データ 1 およびデータ 2 のそれぞれに対して、機械学習を適応し、leave one person out cross-validation で提案手法を評価する。評価するにあたって、利用する機械学習アルゴリズムを検討、選択する。

#### 4.3 アルゴリズムの検討

データをクラスタリングする教師あり機械学習モデルとしては、決定木、ロジスティック回帰、ランダムフォレスト、SVM (カーネル)、ニューラルネットワークなどがある。このうち、決定木やロジスティック回帰は、高速性に優れており、推定の際の計算量も少ない。また、ランダムフォレスト、SVM (カーネル)、ニューラルネットワークなどは、正確性に優れている。さらに、正確性という意味では、深層学習が従来の機械学習アルゴリズムを用いた手法よりも高精度な結果を数多く獲得していることが知られており、単に精度を追い求めるのであれば深層学習を用い

ることも有効と考えられる。一方で、深層学習は、計算コストが大きい。スマートフォンの高性能化にともない、端末上で深層学習モデルを利用することは可能となっているものの、計算量は基本的に大きい。ランダムフォレスト、SVM (カーネル)、ニューラルネットワークなどは、深層学習に比べれば少ないものの、決定木やロジスティック回帰に比べると計算量が大きい。

3章の要件2に対応するため、端末の負荷が少ないシンプルな数理モデルを選択し、推定モデル構築を行う必要がある。そこで、今回は比較的シンプルなモデルとして決定木とロジスティック回帰 (LR) を選択した。また正確性に優れているといわれているアルゴリズムの中で、高速性能も高いといわれているランダムフォレストも合わせて利用し、これら3つの数理モデルに基づいて推定モデルを構築し、比較・評価する。各モデルの特徴は次のとおりである。なお、深層学習については、要件2を満たさないと考え、除外している。

(1) 決定木

決定木は、属性値の条件によりノードを分岐させて木構造の分類器を生成する伝統的な機械学習手法の1つである。決定木学習の特徴として、if-then ルールに基づいて木を生成していくため、他の手法と比べて分類過程の可読性が高いという利点がある。本研究では、決定木学習の代表的なアルゴリズムである C4.5 決定木を用いた。

(2) ロジスティック回帰 (LR)

ロジスティック回帰は、独立変数の値に対する (2 値) 分類結果を 0 または 1 で与えるのではなく、オッズ比と呼ばれる確率として計算するモデルであり、多値分類にも拡張できる。古典的な統計手法であり、計算量も比較的小さい。回帰による分類手法として一般的であるため、ここで使用する。

(3) ランダムフォレスト (RF)

RF はアンサンブル学習を用いた機械学習アルゴリズムの1つである。複数の決定木を弱分類器として使用し、その結果を統合して結果を得る。他の分類器に比べ性能が高く、判定速度も高いといわれているが、決定木やロジスティック回帰に比べると、計算量は一般的に大きい。

5. 評価結果

本章では、手で持つ場合 (Class1~4) と卓上に置く場合 (Class5~8) に、Nexus 5 と Nexus 6 を使用して収集したデータ1とデータ2に対して、3つの機械学習アルゴリズム (決定木, LR, RF) で推定モデルを構築し、推定精度を評価した結果について述べる。

5.1 総合推定結果

すべての組合せにおける操作形態推定結果に対する平均の F 値を表 3 に示す。推定結果について Nexus 5, Nexus 6 に対し、クラス1~4の推定精度を確認する。まず Nexus 5 に関して見てみると、データ1のときは LR の場合が最も高く、61.2%であった。データ2に対しては RF が最も高く、96.5%であった。一方でクラス5~8について見ると、データ1、データ2で高精度だったのはともに RF であり、53.7%, 95.2%であった。

次に Nexus 6 に関してクラス1~4の推定精度を見ると、データ1のときは LR の場合が最も高く、48.9%であった。データ2に対しては RF が最も高く、89.2%であった。一方でクラス5~8について見ると、データ1、データ2で高精度だったのはともに RF であり、52.9%, 85.0%であった。

Nexus 5 と Nexus 6 による精度を比較すると、全体的に Nexus 6 の精度が低くなっていることが確認された。これは表 1 に示したとおり、Nexus 6 の大きさが大きいためと考える。Nexus 6 の画面サイズは 6.0 インチと広がっており、これによりユーザのタッチ操作可能範囲が広がり、操作の自由度が向上したことが精度の低下につながっていると考えられる。また、重さが約 50g 重くなっており、支える手の負担が向上したことも精度の低下につながっていると考えられる。画面サイズや重さによる精度への影響については、今後の検討課題である。

5.2 手で持つ場合での Nexus 5 における操作形態推定結果

5.1 節の結果として、手で持つ場合 (Class1~5) と卓上に置く場合 (Class5~8) の両方で、Nexus 5, および Nexus 6 どちらも、1) データ1よりもデータ2に対して、2) 決定木や LR よりも RF で推定モデルを構築した場合に精度

表 3 総合評価：すべての組合せの F 値一覧

Table 3 Overall evaluation: List of F-values for all combinations.

機種		Nexus 5			Nexus 6		
		決定木	LR	RF	決定木	LR	RF
Class 1~4	データ 1	0.543	0.612	0.576	0.447	0.489	0.479
	データ 2	0.899	0.822	<b>0.965</b>	0.844	0.889	<b>0.892</b>
Class 5~8	データ 1	0.522	0.469	0.537	0.506	0.478	0.529
	データ 2	0.919	0.856	<b>0.952</b>	0.847	0.838	<b>0.850</b>

表 4 データ 1 に対する Nexus 5 での評価結果

Table 4 Evaluation results on Nexus 5 for data 1.

(a) 決定木			
Class	Precision	Recall	F 値
1	0.602	0.606	0.581
2	0.594	0.598	0.584
3	0.608	0.575	0.545
4	0.432	0.434	0.429
Average	0.562	0.563	0.543

(b) LR			
Class	Precision	Recall	F 値
1	0.697	0.678	0.635
2	0.746	0.713	0.601
3	0.670	0.678	0.650
4	0.518	0.499	0.438
Average	0.660	0.644	0.612

(c) RF			
Class	Precision	Recall	F 値
1	0.683	0.692	0.678
2	0.636	0.631	0.614
3	0.682	0.627	0.586
4	0.467	0.463	0.444
Average	0.602	0.596	0.576

が高くなる傾向が見られたため、5.2 節では、手で持つ場合 (Class1~4) における Nexus 5 に絞って詳細な評価結果を示す。

手で持つ場合 (Class1~4) におけるデータ 1, およびデータ 2 に対する Nexus 5 での評価結果を表 4, 表 5 に示す。各表のクラス番号 1, 2, 3, 4 は、図 1 の Class 番号にそれぞれ対応しており、各々のクラスの Precision, Recall, F 値を記載している。

加えて評価に leave one person out cross-validation を採用している。すなわち各被験者データをテストした結果から得られた、Precision, Recall, F 値を単純平均したものを、表中に Average で示している。

データ 1 に対する推定結果

表 4 の (a) 決定木, (b) LR, (c) RF, それぞれに関して平均 Precision, Recall, F 値を見ると、最も良かったのは LR の推定モデルであり、Precision が 66.0%, Recall が 64.4%, F 値が 61.2%であった。

全体的な結果として、スマートフォン操作から得られる特徴量、9 つのみでは精度として不十分な結果となった。LR で構築した推定モデルで、実験参加者 1 名のテストデータから得られた混同行列 (表 6 (a)) を確認すると、右手か左手の識別は 100%推定できるものの、親指もしくはその他の指での操作判定がほとんどできていないため、精度が悪くなったと考えられる。

表 5 データ 2 に対する Nexus 5 での評価結果

Table 5 Evaluation results on Nexus 5 for data 2.

(a) 決定木			
Class	Precision	Recall	F 値
1	0.979	0.979	0.979
2	0.977	0.976	0.976
3	0.991	0.988	0.989
4	0.690	0.736	0.670
Average	0.898	0.923	0.899

(b) LR			
Class	Precision	Recall	F 値
1	0.845	0.768	0.707
2	0.954	0.956	0.954
3	0.988	0.980	0.984
4	0.863	0.775	0.759
Average	0.884	0.853	0.822

(c) RF			
Class	Precision	Recall	F 値
1	0.989	0.989	0.989
2	0.971	0.972	0.971
3	0.995	0.997	0.996
4	0.928	0.887	0.889
Average	0.967	0.969	0.965

表 6 実験参加者 1 名の場合の混同行列

Table 6 Confusion matrix for one participant.

(a) LR					(b) RF				
	1	2	3	4		1	2	3	4
1	14	23	0	1	1	75	0	0	0
2	21	27	0	0	2	0	54	0	0
3	0	0	42	20	3	8	0	138	0
4	0	0	20	29	4	0	0	0	80

データ 2 に対する推定結果

データ 2 に関しても同様に、表 5 の (a) 決定木, (b) LR, (c) RF について順に平均 Precision, Recall, F 値を確認すると、推定結果が最も良かったのは RF の推定モデルで、平均の Precision, Recall, F 値はそれぞれ、96.7%, 96.9%, 96.5%であった。また LR, 決定木の順で推定精度が下がることが分かった。各平均精度の値は LR での推定モデルで、88.4%, 85.3%, 82.2%, 決定木での推定モデルでは、89.8%, 92.3%, 89.9%であった。

全体的な結果として、データ 1 に対する推定結果と比べ飛躍的に推定精度が向上した。最も推定精度が良かった RF で構築した推定モデルから、被験者 1 名のテストデータを用いて導出した混同行列を、表 6 (b) に示す。この被験者は、97.7%と非常に高い精度で推定可能であった。このため、支持手の指の長さは、親指もしくはその他の指での操作判定に有効であることが分かった。また、RF を用いたときの特徴量の寄与度を確認した結果、始点 x 座標、

終点  $x$  座標, 出現最頻領域の  $x$  軸方向の分割番号の寄与度が相対的に高いことが確認された. このことから, 指の長さに加え, スマートフォンの左右方向のどこを使用しているかが, 操作形態の推定に有効な特徴量であることが確認された.

## 6. 考察

本章では, 5章の結果に基づいて, 提案手法の有効性, モデルの比較, 操作形態推定の意義について考察する.

### 6.1 提案手法の有効性

本論文では, スマートフォンの操作形態を推定するための手法を提案した. 3種類の推定モデル (決定木, LR, RF) を用いて比較した結果, F 値約 90%の精度で分類できることを確認している. 今回, 推定に用いたデータは, Yahoo! ニュース閲覧時にスワイプをした際のデータである. そのため, 同一利用者であっても, 大きなスワイプや小さなスワイプ, 早いスワイプやゆっくりなスワイプなど, 様々なスワイプが含まれている. このような状況の中で, 各1回のスワイプのデータから約9割の精度で操作形態を判定できる本手法は有効であると考えられる.

また, 実際に利用する際には, 数回のスワイプに基づいて判定するなど考えられる. たとえば, 3回スワイプデータが得られた時点で, 3回分の推定結果の多数決で判定することなどが考えられる. これにより, 精度はより向上すると考えられ, 実用的な手法として十分な精度が達成できていると考える.

実験から, ユーザの指の長さを考慮することにより高精度な推定が可能となることを確認している. 今回の実験においては, 手作業で手のサイズを測っているが, ユーザの負担を低減するためには, スマートフォンのカメラを用いた計測が有効と考える. 具合的には, Apple の「計測」アプリや Android の「AR Ruler」の機能を用いることで, 手軽に計測可能と考える.

### 6.2 推定モデルの比較

3種類の推定モデルを用いて操作形態の推定精度を比較した結果, Nexus 5ではRFが約95%と他の手法と比べて高い精度を示した. しかしながら, Nexus 6では, Class1~4に対しては約90%, Class5~8に対しては85%となり, モデル間の差はあまり見られなかった. 精度の面だけを見ると, RFを推定モデルとして採用することが望ましい. しかしながら, 端末への負荷や汎用性を考えると検討の余地がある.

決定木の結果をみると, Nexus 5で約90%, Nexus 6で約85%と端末間で違いはあるものの, RFに次いで良い結果となっている. また, 計算量はRFよりも少ないことは明らかである. 3章で述べた要件2を考慮すると, 一方の

端末 (Nexus 5) の精度を5%向上させるためだけに, 計算量すなわち消費電力の高いモデルを採用することは望ましいとはいえない. 決定木のようなシンプルなモデルで判定できる推定モデルの採用も考えられる. また, 同サイズの他端末への移植容易性の面からは, 決定木のように可読性の高いモデルのほうが適していると考えられる.

実際のスマートフォンへの実装には, これらの点も考慮して, 推定モデルを選択する必要がある. 計算量 (消費電力) や汎用性と精度とのトレードオフについては, 今後の検討課題である.

### 6.3 提案手法の適用範囲と限界

はじめに, 本研究で定めた操作形態の定義について限界を述べる. 本論文では, 操作頻度の多い8つの基本的な操作形態を対象とし, 約90%の精度で推定できることを確認している. 利用頻度の高い操作形態を一定以上の精度で推定できるという点で提案手法は実利用に適用可能と考える. 一方で, 操作形態の定義において, 操作する指を親指と人差し指に限定しているが, 中指, 薬指, 小指を主に使用する人もいることが確認されている. さらに, ユーザのコンテキストによって, 操作する指は変化する. たとえば, 喫煙時であれば, 左でスマートフォンを支え, 右手で煙草を持ちながら薬指や小指で操作することなどが考えられる. このことを考えると, 本論文で対象とした8種類以外の操作形態も推定可能にする必要があり, 限界といえる.

次に, 実験設定の限界について述べる. 本研究での学習データは, Yahoo! ニュースアプリ閲覧時の操作データのみを利用しているが, 他のアプリでもタッチやスワイプ操作は多く使用されているため, 提案手法は他のアプリ操作時においても適用可能と考える. しかしながら, ゲームアプリや地図アプリなど, マルチタッチやピンチなどが必要なアプリにただちに適用することはできない. 先に述べたとおり, 今後対象とする操作形態の定義を拡張し, より多くの操作形態を推定可能にする必要がある. また, 今回の実験対象者は20代に限定されている. 被験者の年代により操作スキルが異なると考えられるため, 今後, 多様なアプリ, 多様な年代での検証が必要と考える.

最後に, TouchAnalyzerの適用可能範囲について述べる. TouchAnalyzerは, 少なくとも (Galaxy S II, Nexus 5, Nexus 6) に対し対応可能なことを確認している. この他の端末についても, イベントログの形式や出力ファイルへのパス違いを確認し吸収していくことで, 多様な端末に対応可能と考えている. ただし, 最新端末や root 取得ができない端末などに対しては適応不可能であり, 本研究の限界であると考えられる. この問題の解決方法としては, TouchAnalyzerをスマートフォン端末に初期インストールされているネイティブアプリとして採用することが考えられる.

## 6.4 操作形態推定の意義

本研究は、タッチ操作ログをセンサと見なし、ユーザのコンテキストを推定するという研究の一環である。本論文では、コンテキストの1つとして、操作形態に着目し、その推定手法の提案と評価を行った。スマートフォンの操作形態を整理し、特徴量から操作形態を一定精度で認識できるという結果を示した。これらの知見は、最新端末でのアプリ作成や OS 開発に対する知見になると考える。1章で述べたように、操作形態が分かることによって、ユーザのスキルに応じた UI を提供したり、持ち方に応じて誤操作を低減するといったことが可能となる。今回の結果から、高い精度で操作形態を把握できることが明らかになったことで、より幅広いコンテキストの推定にタッチ操作ログを利用できる可能性が高まったといえる。たとえば、同一ユーザであっても、状況に応じてタッチ操作形態が変化することを利用し、何をしながらスマートフォンを操作しているのかという行動認識への応用や、スワイプの速度やタッチの強さから焦りを認識するシステムなどが考えられる。

今回はスマートフォンを対象としたが、最近では、Pepper などの対話型ロボット、サイネージ、自動車など、生活のあらゆるところにタッチパネルが広がってきている。機器へのタッチ操作を通じ、機器側が操作する人の状態を認識することができるようになれば、これまでよりも一層コンテキストアウェアなシステムを実現できる可能性があり、本研究の成果は非常に意義深いものであると考えている。

## 7. まとめ

本論文では、タッチ操作ログから、定義した8種類の操作形態（支持している手、操作している手、操作している指の組）を推定する手法を提案した。具体的には、TouchAnalyzer から得られた情報を特徴量として、機械学習手法により操作形態を推定するための手法を構築した。2種類のデータ、3種類の機械学習アルゴリズム、2種類のデバイスを使用し、それぞれの推定精度を評価した。その結果、指の長さが操作形態の推定に有効な特徴量であること、RF が操作形態の推定に有効（F 値約 90%）であること、スマートフォンのサイズによって推定精度が変化すること、を明らかにした。

より多くの操作形態を対象にすること、実験対象を多様なアプリ・年代に拡大することなどが今後の課題である。

謝辞 本研究は、JSPS 科研費 (26700007) の助成を受けたものである。

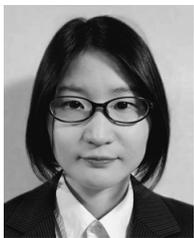
## 参考文献

[1] Ho, J. and Intille, S.S.: Using context-aware computing to reduce the perceived burden of interruptions from mobile devices, *Proc. SIGCHI Conference on Human Factors in Computing Systems*, pp.909-918, ACM (2005).  
 [2] Partridge, K. and Begole, B.: Activity-based advertising

techniques and challenges, *Proc. Workshop on Pervasive Advertising* (2009).  
 [3] FreakOut: フリークアウトと日本交通、位置情報連動マーケティング事業で提携 日本最大の配車アプリ「全国タクシー」の位置情報をデジタル広告配信に活用, 入手先 (<https://www.fout.co.jp/pr20160107/>).  
 [4] Goel, M., Wobbrock, J. and Patel, S.: GripSense: Using Built-in Sensors to Detect Hand Posture and Pressure on Commodity Mobile Phones, *Proc. 25th Annual ACM Symposium on User Interface Software and Technology, UIST '12*, pp.545-554, ACM (online), DOI: 10.1145/2380116.2380184 (2012).  
 [5] Harrison, B.L., Fishkin, K.P., Gujar, A., Mochon, C. and Want, R.: Squeeze Me, Hold Me, Tilt Me! An Exploration of Manipulative User Interfaces, *Proc. SIGCHI Conference on Human Factors in Computing Systems, CHI '98*, pp.17-24, ACM Press/Addison-Wesley Publishing Co. (online), DOI: 10.1145/274644.274647 (1998).  
 [6] Kim, K.-E., Chang, W., Cho, S.-J., Shim, J., Lee, H., Park, J., Lee, Y. and Kim, S.: Hand Grip Pattern Recognition for Mobile User Interfaces, *Proc. 18th Conference on Innovative Applications of Artificial Intelligence - Volume 2, IAAI'06*, pp.1789-1794, AAAI Press (2006) (online), available from (<http://dl.acm.org/citation.cfm?id=1597122.1597138>).  
 [7] 塚本昌克, 樋口雄太, 岡田 隆: 把持特徴を利用したユーザインタフェースの開発: Grip UI, NTT DoCoMo テクニカル・ジャーナル, Vol.21, No.3, pp.17-23 (2013).  
 [8] Taylor, B.T. and Bove, Jr., V.M.: Graspables: Grasp-recognition As a User Interface, *Proc. SIGCHI Conference on Human Factors in Computing Systems, CHI '09*, pp.917-926, ACM (online), DOI: 10.1145/1518701.1518842 (2009).  
 [9] Cheng, L.-P., Lee, M.H., Wu, C.-Y., Hsiao, F.-I., Liu, Y.-T., Liang, H.-S., Chiu, Y.-C., Lee, M.-S. and Chen, M.Y.: IrotateGrasp: Automatic Screen Rotation Based on Grasp of Mobile Devices, *Proc. SIGCHI Conference on Human Factors in Computing Systems, CHI '13*, pp.3051-3054, ACM (online), DOI: 10.1145/2470654.2481424 (2013).  
 [10] Buschek, D. and Alt, F.: TouchML: A Machine Learning Toolkit for Modeling Spatial Touch Targeting Behaviour, *Proc. 20th International Conference on Intelligent User Interfaces, IUI '15*, pp.110-114, ACM (online), DOI: 10.1145/2678025.2701381 (2015).  
 [11] Goel, M., Jansen, A., Mandel, T., Patel, S.N. and Wobbrock, J.O.: ContextType: Using Hand Posture Information to Improve Mobile Touch Screen Text Entry, *Proc. SIGCHI Conference on Human Factors in Computing Systems, CHI '13*, pp.2795-2798, ACM (online), DOI: 10.1145/2470654.2481386 (2013).  
 [12] 梶田美帆, 阿部和樹, 中村聡史, 山中祥太: 親指の長さがスマートフォンの操作に及ぼす影響の調査と親指の長さの自動推定手法の検討, 情報処理学会研究会報告ヒューマンコンピュータインタラクション, HCI, Vol.2019-HCI-185, No.20, pp.1-8 (2019).  
 [13] Löchtefeld, M., Schardt, P., Krüger, A. and Boring, S.: Detecting users handedness for ergonomic adaptation of mobile user interfaces, *Proc. 14th International Conference on Mobile and Ubiquitous Multimedia*, pp.245-249, ACM (2015).  
 [14] Buschek, D., Rogers, S. and Murray-Smith, R.: User-specific Touch Models in a Cross-device Context, *Proc. 15th International Conference on Human-computer Interaction with Mobile Devices and Services, MobileHCI '13*, pp.382-391, ACM (online), DOI: 10.1145/2493190.

2493206 (2013).

- [15] Hirabe, Y., Arakawa, Y. and Yasumoto, K.: Logging All the Touch Operations on Android, *The 7th International Conference on Mobile Computing and Ubiquitous Networking (ICMU 2014)*, pp.93-94 (2014).
- [16] 平部裕子, 津田麻衣, 荒川 豊, 安本慶一: タッチ操作の挙動に基づくユーザプロファイル推定手法の提案, 情報処理学会研究報告, Vol.2014-UBI-41, No.20, pp.1-6 (2014-03).
- [17] 平部裕子, 荒川 豊, 安本慶一: TouchContext: タッチ操作の挙動分析に基づく人のコンテキスト認識, 情報処理学会, インタラクション 2014, インタラクティブ発表, No.A0-5, pp.158-165 (2014).
- [18] Hirabe, Y., Suwa, H., Arakawa, Y. and Yasumoto, K.: TouchAnalyzer: A System for Analyzing User's Touch Behavior on a Smartphone, *International Journal of Computer Science and Mobile Computing*, Vol.7, pp.25-38 (2018).
- [19] Steven Hooper, How Do Users Really Hold Mobile Devices?, available from (<https://www.uxmatters.com/mt/archives/2013/02/how-do-users-really-hold-mobile-devices.php>) (accessed 2020-07-21).
- [20] orefolder.net, 【アンケート企画】あなたのスマホの持ち方教えてください！ 2017【結果と考察編】, 入手先 (<https://www.orefolder.net/blog/2017/07/hands-enquete-result-2017/>) (参照 2020-07-21).
- [21] 船田哲男: 推定された声帯音源波の個人性評価と類似性判断, 日本音響学会誌, Vol.43, No.6, pp.389-395 (1987).
- [22] 河内まき子: 2012: AIST 日本人の手の寸法データ, 入手先 (<https://www.airc.aist.go.jp/dhrt/hand/index.html>) (参照 2020-07-21).



平部 裕子 (正会員)

2013年鹿児島大学工学科卒業。2015年奈良先端科学技術大学院大学情報科学研究科博士前期課程修了。2018年奈良先端科学技術大学院大学情報科学研究科博士後期課程修了。博士(工学)。2018年シャープ株式会社入社。

2021年から三菱重工業株式会社 ICT ソリューション本部 EPI 部社員。



諏訪 博彦 (正会員)

2006年電気通信大学大学院情報システム学研究科博士後期課程修了。博士(学術)。電気通信大学助手・助教・研究員, 奈良先端科学技術大学院大学助教・特任准教授, 理化学研究所研究員を経て, 2021年11月より奈良先端科学技術大学院大学准教授。



荒川 豊 (正会員)

2001年慶應義塾大学理工学部卒業。2006年同大学院博士課程修了。博士(工学)。同大学助教, 九州大学助教, 奈良先端科学技術大学院大学准教授を経て, 2019年より九州大学大学院システム情報科学研究院教授。ユビキタスコンピューティングに関する研究に従事。



安本 慶一 (正会員)

1991年大阪大学基礎工学部情報工学科卒業。1995年同大学大学院博士後期課程退学。2011年より奈良先端科学技術大学院大学情報科学研究科教授。ユビキタスコンピューティングに関する研究に従事。