

NAIST-IS-MTDD1561017

博士論文

情報通知タイミングに関わるコンテキスト推定のための
タッチ操作解析およびグループ検出に関する研究

平部 裕子

2018年3月15日

奈良先端科学技術大学院大学
情報科学研究科 情報科学専攻

本論文は奈良先端科学技術大学院大学情報科学研究科に
博士（工学）授与の要件として提出した博士論文である。

平部 裕子

審査委員：

安本 慶一 教授 （主指導教員）

中村 哲 教授 （副指導教員）

荒川 豊 准教授 （副指導教員）

諏訪 博彦 助教 （副指導教員）

藤本 まなと 助教 （副指導教員）

情報通知タイミングに関わるコンテキスト推定のための タッチ操作解析およびグループ検出に関する研究*

平部 裕子

内容梗概

スマートフォンの急速な浸透を背景に、スマートフォンのセンサデータを活用して、最適な情報を、ユーザの注目を引きやすいタイミングで通知する、情報通知技術に関する需要が高まっている。この実現には、(1) 対象者の注目を引きやすいタイミングを人の状況（コンテキスト）から推定し、(2) 誰といるかを検出し最適な通知内容を判定する必要がある。(1) に関して、従来研究では、加速度、ジャイロセンサを利用し、立つ、座る、走る、歩くななどの単純な動作に対するコンテキスト推定が多くなされているが、これら単純な動作に紐付いた通知では、最適な通知内容を選択するのが難しいという問題がある。(2) に関して、位置情報や移動情報を利用して、最寄りや評価の高い飲食店を推薦したり、電車やバスの乗車に最適な時刻をユーザに知らせるサービスが提供されている。これらの通知内容はサービス利用者にとって便利であるものの、同伴者を考慮したものは存在しない。

本研究では、情報通知タイミングに関わるコンテキスト推定を可能にする基盤システムの実現を目指し、汎用性を考慮してスマートフォン単一で実現可能な方法として、タッチパネルへの入力（タッチ操作）解析および BLE (Bluetooth Low Energy) を用いたグループ推定に関する手法を提案する。タッチ操作からコンテキスト推定を可能にするため、タッチ操作挙動を統計情報へと変換し、分析可能とするシステムを開発した。本システムは OS が出力する低水準な情報を逆解析し、シングル（マルチ）タッチ、シングル（マルチ）スワイプ、ピンチイン（アウト）、ローテートの検出を可能にする。同時に、スワイプ速度やタッチ挙動頻度、接地面積、圧力といった統計情報も取得する。また本システムを利用し、タッチ操作から得られるコンテキストの一つである、スマートフォンの操作形態（持ち方）推定を

*奈良先端科学技術大学院大学 情報科学研究科 情報科学専攻 博士論文, NAIST-IS-MTDD1561017, 2018年3月15日.

行った．Android 端末上で一般的な 8 種類の操作形態を推定する機械学習アルゴリズムを開発し，16 名の被験者で評価した結果，概ね 90% の確率で推定可能な事を確認した．

(2) に関して，BLE によるグループ検出を可能にするため，スマートフォンに搭載された BLE の RSSI 値を取得するシステムを開発した．開発システムを利用して，BLE 伝播特性モデルを作成し，実験的に最適な閾値を決定したうえで，グループ検出可能かをシナリオベースの実験により確認したところ，結果として，正確にグループ検出できることがわかった．

キーワード

コンテキスト推定, タッチ操作分析, 人間関係, Android, BLE, 割り込み通知

A Study on Smartphone Touch Operation Analysis and Human Group Detection toward Timely Information Notifications*

Yuko Hirabe

Abstract

Recent rapid penetration of smartphones has brought increasing demands for realizing the information notification technology which enables optimal information notification in terms of both timings and contents utilizing sensors embedded in a smartphone. To realize the information notification technology, two primary techniques are needed: (1) estimating suitable timings for easy attention drawing from the user's context; (2) inferring the companions who are with the user and deciding optimal contents of notifications to the user.

In the previous research regarding (1), there are many methods to infer contexts to simple motions, such as standing, sitting, running, and walking, by using accelerometer and gyroscope. However, in the notifications based on the simple motions, it is difficult to select suitable notified-contents. Regarding (2), by harnessing information of locations and movements of the user, systems of the recommendation and service that let a user know useful information (high rating-restaurant nearby the user or optimum time for the user to get on a bus or a train) exists. The information notified to the people who use these systems is helpful. However, there is no notification system for recommendation and service that considers companions of the people.

In this thesis, the main aim is to realize a basic system to enable estimation of

*Doctoral Dissertation, Graduate School of Information Science,
Nara Institute of Science and Technology, NAIST-IS-DD1561017, March 15, 2018.

user's contexts concerning the timing of information notification. Considering universal use in the future, we proposed methods of analysis to users' input (touch operation) to touch panel on smartphones, also group estimation by harnessing BLE (Bluetooth Low Energy) advertisement signals provided on cell phones.

To be able to estimate user contexts based on touch operations, we developed a system, TouchAnalyzer, which allows us to analyze touch operations of users by transforming to statistical information. TouchAnalyzer decodes low-level data outputted by OS (Operation System), and can detect touch operations like single-(multi-) touch and swipe, pinch-in(-out), and rotate. In addition, statistical information (swipe speed, the frequency of touch operation, area size touched and pressured by a user) are obtained on TouchAnalyzer. Utilizing the developed system, we tried to estimate user's hand-holding of a smartphone, which is one of the contexts acquired from analysis of touch operations information. We employed machine learning algorithms to predict eight hand-holdings defined on Android in general, and estimated the eight hand-holdings by using the algorithms with the data collected from 16 participants. As a result, the algorithms achieved around 90% mean F-values as estimation accuracy.

About (2), we developed a system for the acquisition of RSSI (Received Signal Strength Indicator) values of BLE signals on a smartphone to realize group detection. Applying the RSSI acquirement system, we created a model of propagation characteristic of RSSI. On the situation selected as the most appropriate thresholds experimentally, the model was evaluated in the scenario-based experiments and we confirmed that group detection can be done almost precisely with the model.

Keywords:

context estimation, touch operation analysis, human proximity, Android, BLE, interference notification

目次

図目次		viii
表目次		1
第 1 章	序論	2
1.1	研究背景	2
1.2	研究目的	4
第 2 章	タッチ操作コンテキストを利用した情報通知タイミング推定に向けた基盤技術	7
2.1	あらまし	7
2.2	スマートフォンのタッチ操作情報取得手法	11
2.2.1	関連研究	12
2.2.2	Android の一般的なタッチ操作取得経路	13
2.2.3	タッチ情報取得方法の検討	14
2.2.4	タッチ操作を取得する上での技術的課題	15
2.2.5	TouchAnalyzer の設計と実装	16
	TouchAnalyzer の全体概要	16
2.2.6	タッチ操作の認識	20
	シングルタッチ・マルチタッチ	21
	シングルスワイプ・マルチスワイプ	22
	ピンチ	22
	ローテート	22
2.2.7	タッチ操作の統計分析	23
2.2.8	タッチ操作の可視化	25
2.2.9	TouchAnalyzer 評価実験および評価方法	26
	結果および考察	27
2.3	タッチ操作分析に基づく操作形態推定手法	29
2.3.1	関連研究	30

	組み込みセンサや追加センサを利用した推定手法	30
	機械学習を用いた手法	30
	本研究の狙い	31
2.3.2	操作形態の定義	31
2.3.3	要件設定	33
2.3.4	推定プロセスの全体概要	34
2.3.5	操作形態推定手法の実現	35
	実験の環境と内容	36
	特徴量	36
	アルゴリズムの検討	37
2.3.6	結果	38
	Nexus 5 における操作形態推定結果	40
	総合推定結果	42
2.3.7	考察	43
	提案手法の有効性	43
	推定モデルの比較	44
	操作形態推定の意義	45
2.4	まとめ	46
第 3 章	BLE を利用したグループ検出手法	47
3.1	あらまし	47
3.2	関連研究	49
	3.2.1 グループ推定の既存システムおよび研究	50
	3.2.2 既存のグループ推定手法の課題	51
	3.2.3 要件設定	52
3.3	システム概要	53
	3.3.1 BLE 計測アプリケーション	55
	3.3.2 グループ検出手法	57
	3.3.3 本システムの利点	57
3.4	評価実験	57
	3.4.1 実験の概要	58

3.4.2	BLE 信号の伝播特性	59
3.4.3	人の近接度に基づくグループ検出への効果	60
3.4.4	シナリオに基づく提案手法の評価実験	62
3.4.5	シナリオに基づくグループ検出率	64
3.4.6	考察	65
3.5	まとめ	67
第 4 章	結論	68
	参考文献	72
	業績リスト	78

目次

1.1	最適な情報通知タイミングで実現できるインテリジェントな通知の例 *	3
1.2	本研究の目的, 必要技術, 研究対象部分の関連性	5
2.1	適切な情報通知タイミングとタッチコンテキストの関連	8
2.2	通知に良いあるいは良くないタイミングの定義	9
2.3	通知タイミングとタッチ操作コンテキスト推定を実現するための必要技術	10
2.4	タッチ操作の認識の流れと提案システムとの関連	14
2.5	TouchAnalyzer の全体概要	16
2.6	出力されるログデータの 1 文	18
2.7	シングル・マルチタッチ	20
2.8	シングル・マルチスワイプ	20
2.9	ピンチ	21
2.10	ローテート	21
2.11	ピンチとローテートの定義	23
2.12	タッチ操作の挙動に見られる軌跡	24
2.13	ヒートマップによるタッチ操作の可視化	25
2.14	アプリケーションの実験画面	26
2.15	人の詳細状況により変化する持ち方 (操作形態) の例	29
2.16	操作形態の種類	32
2.17	推定プロセス	35
3.1	適切な情報通知タイミングと同伴者コンテキストの関連	48
3.2	同伴者コンテキスト推定への必要技術	48
3.3	近接度 (人々の近さ) とグループの定義	49
3.4	システム概要図	54
3.5	開発アプリケーション概要図	55

3.6	開発アプリケーションの画面構成	56
3.7	実験シナリオの全体像	58
3.8	端末間距離と RSSI 値の関係性	59
3.9	4 つの RSSI 値での閾値に依存した人の近接図	61
3.10	シナリオ (1) における人の近接図	63
3.11	シナリオ (1)' における人の近接図	63
3.12	シナリオ (2) における人の近接図	63
3.13	シナリオ (4) における人の近接図	64

表目次

2.1	Android 端末および Android OS ごとにタッチイベントログのフォルダパスの違い	17
2.2	2 点タッチした際に出力されたログ	19
2.3	評価用アプリケーション上でのタッチ操作挙動の認識精度	27
2.4	指の違いによるシングルスワイプの認識精度の違い	28
2.5	データ 1 に対する nexus 5 での評価結果	39
2.6	データ 2 に対する nexus 5 での評価結果	40
2.7	実験参加者 1 名の場合の混同行列	42
2.8	総合評価：すべての組み合わせの F 値一覧	43
3.1	機能比較表	53
3.2	シナリオごとのグループ検出率	65

第 1 章 序論

1.1 研究背景

スマートフォンやウェアラブルデバイスを筆頭とする IoT 機器の急速な浸透を背景に、人は様々なアプリケーションから、日々数多くの通知を頻繁に受け取っている。例えば、メールや LINE*などのコミュニケーションサービスでは、ユーザ間で必要な情報の素早い遣り取りに対して通知が有用であり、Yahoo!ニュース[†]や SmartNews[‡]などのニュースアプリは、ユーザに最新の情報を迅速に通知可能である。これらの通知によって、端末利用者はアプリからの情報を網羅的に取得できる。しかしながら、頻繁な情報通知は、ユーザの集中力を低下させ、ストレスを引き起こすといった問題が指摘されている [1, 2]。この問題を改善するため、普及率の観点から IoT 機器の中でも、特にスマートフォンのセンサデータを活用し、ユーザが注目しやすいタイミングを推定して、最適な情報を通知する情報通知タイミング選定技術の需要が高まっている [3, 4, 5]。このような情報通知タイミング選定技術を実現できれば、図 1.1 に示すような、これまで以上にインテリジェントな通知 (以後、インテリジェント通知) を実現できる。

インテリジェント通知の一例として、端末利用者のスマートフォン操作コンテキストに応じた情報通知タイミング制御が想定される。例えば、ユーザがスマートフォン操作時にあまり集中していない状況 (喫煙中や行列待ち等) では、通知しても良いタイミングであるため、情報を通知する。その一方で、集中している状況 (ゲーム中や仕事中等) では、通知に良くないタイミングであるため、カレンダーや Facebook[§]アプリケーションなどの通知を行わない情報通知のタイミング選択が実現できる。また、端末利用者が日頃から誰と共にいるのか (以後、同伴者コンテキスト) がわかれば、同伴者が上司か同僚か、もしくは初対面かなどを推定し、通知の有無を判定する情報通知タイミングの制御が実現できる。

さらに、上記で推定したスマートフォン操作コンテキストや同伴者コンテキスト

*LINE: <https://line.me/en/>

[†]Yahoo!ニュース: <https://news.yahoo.co.jp/>

[‡]や SmartNews: <https://www.smartnews.com/ja/>

[§]Facebook: <https://www.facebook.com/>

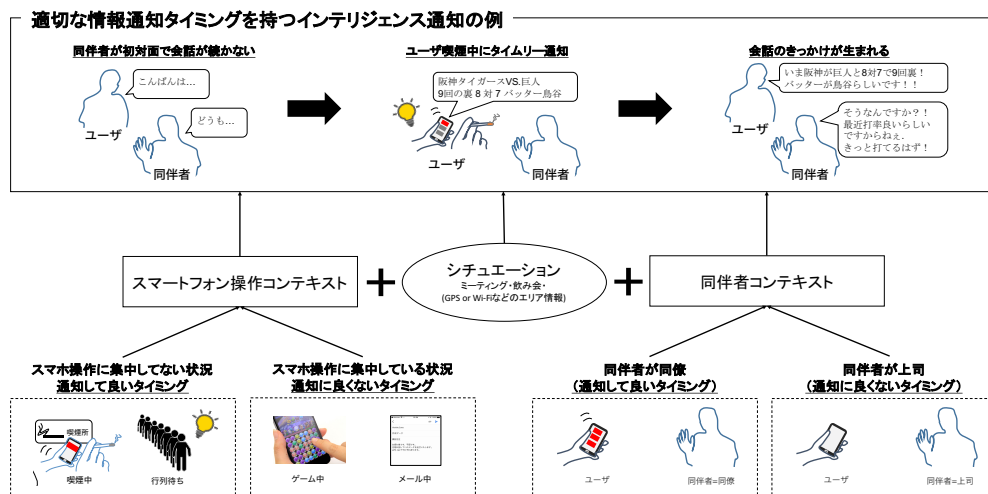


図 1.1: 最適な情報通知タイミングで実現できるインテリジェントな通知の例[¶]

の情報に、GPS (Global Positioning System) や Wi-Fi から得られるエリア情報 (例：職場や居酒屋、学会会場など) に紐づく大まかなシチュエーションを組み込むことで、情報通知タイミングを改善でき、よりインテリジェントな通知を実現できる可能性は高い。例えば、居酒屋での飲み会で、初対面の同僚同士が緊張し、共通の話題が見つからず何も話さない (あるいは話が弾まない) まま、喫煙や食事をしながらスマートフォンを操作して時間を持て余す状況は、しばしば見られる光景である。このような状況において、ニュースや野球の試合速報、初対面の人たち向けの簡単なレクリエーションゲーム情報などを、スマートフォンが適切なタイミングでタイムリーに通知できれば、会話のきっかけを作ることができ、会話の促進や円滑な人間関係の構築にまで繋げることができる。

このようなインテリジェント通知の実現には、情報通知タイミング選定技術の確立が必須であり、以下の 2 つの推定技術を確立する必要がある。

- (1) 対象者の注目を引きやすいタイミングを人の状況 (コンテキスト) から推定
- (2) 誰といるか (同伴者コンテキスト) の推定

(1) に関して、従来研究では、加速度やジャイロセンサを利用し、立つ、座る、

[¶] ゲーム中の図を下記サイトより引用

APPBANK: <http://www.appbank.net/2015/10/14/goods-books/1109444.php>

走る，歩くなどの単純な動作に対するコンテキスト推定 [8, 9, 10] が多くなされているが，日常生活で頻繁に行われるこれらの単純な動作に紐付いた通知では，ユーザが絶えず情報を受け取るため，煩わしさを感じる可能性が高い．情報通知タイミングを最適化するには，単純な動作だけではなく，詳細なスマートフォン操作コンテキスト（喫煙中，ゲーム中，工作中，飲酒中など）の推定が重要になると考えられる．しかしながら，スマートフォンから得られる情報を用いてスマートフォン操作コンテキストを推定している研究はほとんど存在しない．スマートフォン操作コンテキストを推定するには，タッチパネルへの入力情報（タッチ操作）を，継続的に取得し，統計的な情報として分析する必要がある．

（２）に関して，SHARP^{||}から発売されている AQUOS 端末エモパー^{**}では，ユーザが電車やバスに乗車するタイミングに合わせて通知を行なうサービスが展開されている．また通知ではないが，Google^{††}が提供する地図アプリケーション^{‡‡}では，位置情報や移動情報を利用し最寄り度で評価の高い飲食店を表示する機能を有している．これらの通知や情報提供はサービス利用者にとって便利であるものの，同伴者によっては必ずしも適切な情報に成るとは言い難い．また，従来研究において，同伴者を考慮して情報通知タイミングを検出しているものは，ほとんど存在しない．したがって，同伴者の情報を取得するには，構成されるグループを正確に検出し推定する必要がある．

1.2 研究目的

図 1.2 に，本研究の目的，必要技術，ならびに研究対象部分の関連性を示す．本研究では，インテリジェント通知を実現するため，必要な情報通知タイミング選定技術の確立を目的とする．本目的を達成するには，通知タイミングの最適化が求められる．通知タイミングを最適化するには，人の詳細情報の推定が重要であり，（１）対象者の注目を引きやすいタイミングを人の状況（コンテキスト）推定と，（２）誰といるか（同伴者コンテキスト）の推定，が必要となる．本研究では（１），（２）を

^{||}SHARP: <http://www.sharp.co.jp/>

^{**}エモパー: <http://k-tai.sharp.co.jp/dash/emopa/>

^{††}Google: <https://www.google.com>

^{‡‡}Google Map: <https://www.google.co.jp/maps>

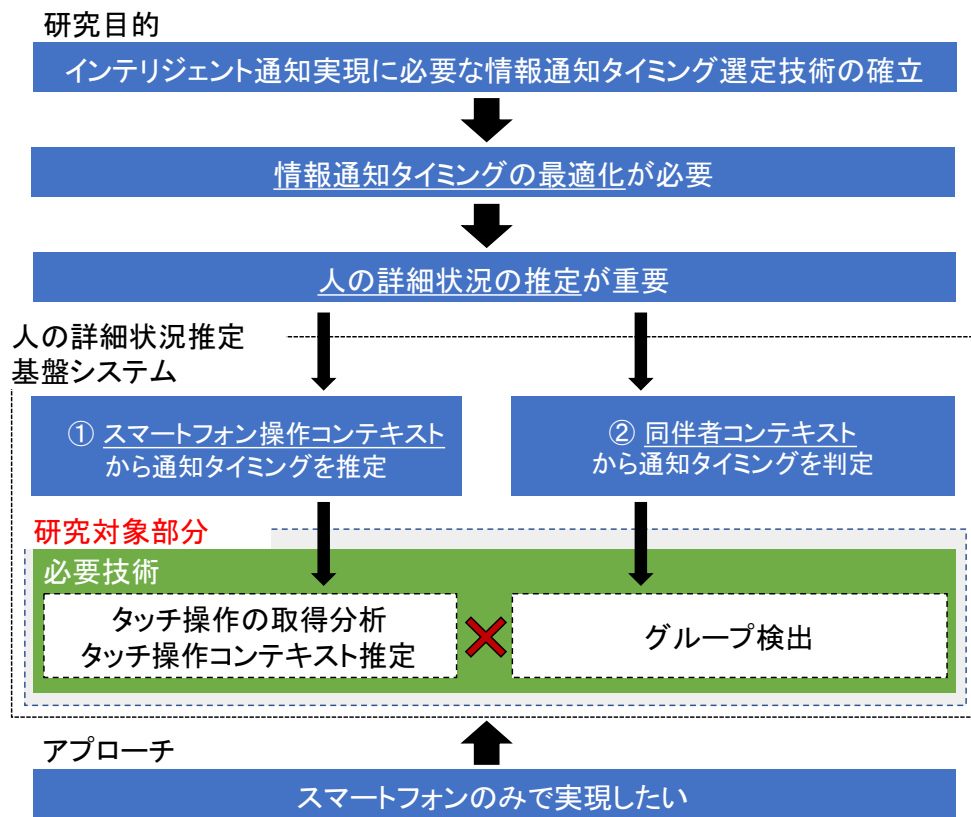


図 1.2: 本研究の目的，必要技術，研究対象部分の関連性

実現する基盤手法の構築を目標として，タッチ操作解析手法および BLE を用いたグループ推定に関する手法を新たに提案する．本論文では，以下の 4 章から構成される．

第 2 章では，タッチ操作解析について詳述する．具体的には，まずはじめに，タッチ操作からコンテキストを推定するため必要な統計情報を取得分析するシステム (TouchAnalyzer) について述べる．本システムは OS が出力している，人間には認識し難い情報を解析することで，シングル (マルチ) タッチ，シングル (マルチ) スワイプ，ピンチイン (アウト)，ローテートの検出を可能とし，同時に，スワイプ速度やタッチ挙動頻度，接地面積，圧力といった統計情報も取得できる．また本システムはタッチパネル上で触れられた場所を，点やヒートマップで可視化できる．その後，TouchAnalyzer を利用し，タッチ操作から得られるコンテキストと

して、基本的なスマートフォンの操作形態（持ち方）の推定手法について述べる．Android 端末上で一般的な 8 つの操作形態を、機械学習を用いて推定するアルゴリズムを開発し、16 名の被験者データを leave one person out 交差検証で評価した結果、概ね 90% の確率で推定可能な事を確認した．

第 3 章では、BLE によるグループ推定を可能にするため、スマートフォンに搭載された BLE の RSSI 値を取得するシステムの開発について述べる．開発システムを利用して、BLE 伝播特性モデルを作成し、実験的に最適なしきい値を決定したうえで、グループ検出可能かをシナリオベースの実験により確認した．結果として、正確にグループ検出できることがわかった．

第 4 章は、結論であり、本研究で得られた結果をまとめる．

第2章 タッチ操作コンテキストを利用した情報通知タイミング推定に向けた基盤技術

本章では、適切な情報通知タイミング推定に向けてタッチ操作コンテキストを利用する事を提案し、情報通知タイミングとタッチ操作コンテキストの関連について説明する。それを踏まえ、タッチ操作コンテキストから通知タイミングを推定するための必要技術について述べた後、各必要技術、タッチ操作取得・分析システム (TouchAnalyzer)、並びに操作形態推定手法に関して詳述する。

2.1 あらまし

従来通知では、適切なユーザ状況（通知タイミング）を考慮せず通知が行われており、端末利用者にとって適切なタイミングで通知されているとは言い難い。例えば図 2.1 に示すように、人は1日の行動の中で単純な動作だけでも、立つ、座る、歩く、走るなど、様々な状態をとっている。単純な動作の中でも端末操作に集中できない動作状況（例：走っている、歩いている、階段の上り降り）がある。このような動作状況では、ユーザが通知に気を取られることで人や障害物と接触し、転倒して怪我をしてしまったり、接触した相手に怪我を負わせてしまう可能性もある。このような動作状況、いわゆる“歩きスマホ*”は、想定される代表的な事例であり、危険な動作状況と認識されている。そのため、ユーザが歩きスマホを行なう場合はアラートを行う手法が提案されていたり [6]、Duolingo[†]と呼ばれる無料の外国語学習プラットフォームシステムでは、反復学習をユーザへ促すため情報通知を行なうが、対象者が歩いている場合は通知しないようデザインされている[‡]等、対策が講じられている。このように端末操作に集中できない動作状況では、情報通知がなされないことが望ましい。

しかしながら、歩く以外の動作に対して、従来の通知では端末利用者の状況に関

*歩きながらスマートフォンを操作する動作状況

[†]Duolingo: <https://ja.duolingo.com/>

[‡]Adobe Blog: <https://theblog.adobe.com/notification-overload-best-practices-for-\designing-notifications-with-respect-for-users/>

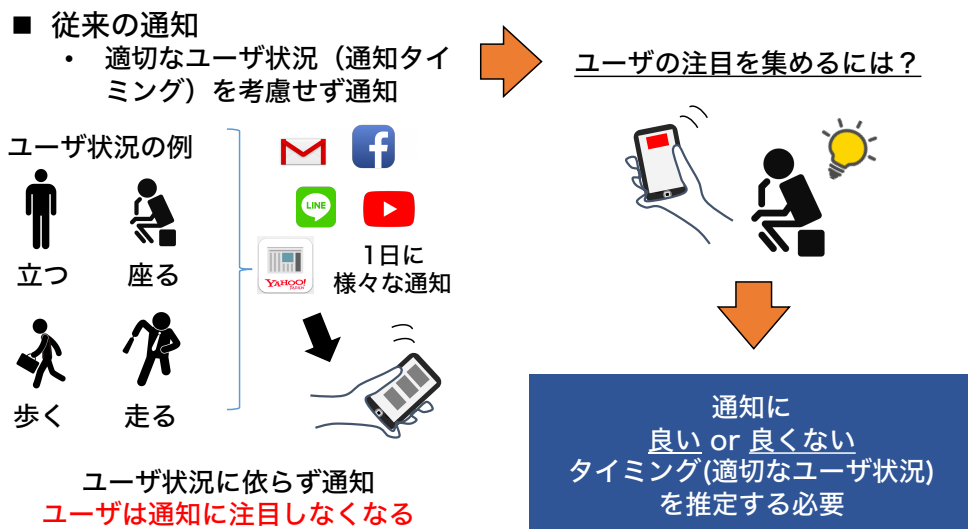


図 2.1: 適切な情報通知タイミングとタッチコンテキストの関連

ならず，無数のアプリケーションから絶えず通知が行われている．頻繁な情報通知によって，端末保持者が通知に注目しなくなる可能性は高い．実際に Abhinav らは，35 名の被験者から 70,000 の通知に対するユーザ反応を収集しアプリケーションと紐付けて調査した結果，家族などと連絡するチャットアプリケーション比べると，ほとんどのアプリケーション通知が無視されている事を明らかにしている [7]．

端末利用者の注目を集めるには，適切なユーザ状況（通知に良い，あるいは良くないタイミング）を推定する必要がある．通知に良い，あるいは良くないタイミングとは，ユーザが通知に注目するため端末を見ていて，かつスマートフォン操作に集中しているかどうかであり，このタイミングによってユーザの通知への注目が変化することが考えられる．例えば，端末保持者が，ゲームなどのアプリケーションを利用している際に通知が来た場合，ほとんどの通知は邪魔であり，無視されていると想定される．実際に多くのユーザがゲーム中の通知を邪魔ととらえて様々な対応を取っている．具体的には，ゲーム利用中のみコミュニケーションアプリやメールの通知を手動でオフにしていたり，アプリ動作をコントロール可能にする，

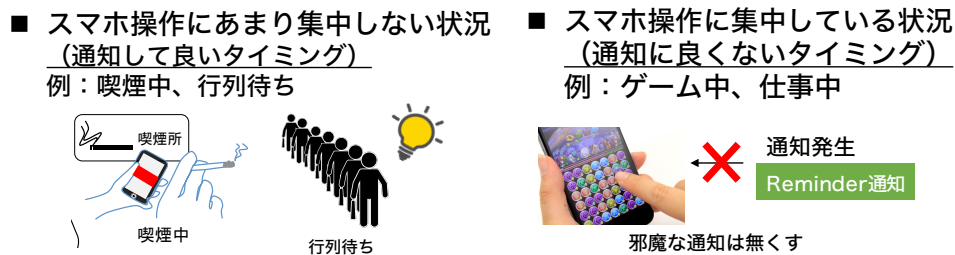


図 2.2: 通知に良いあるいは良くないタイミングの定義

Tasker[§]や SecureTask[¶]と呼ばれるアプリケーションを利用して、スマートフォンの状態に紐づきシステム動作をカスタム設定（例：イヤホンをスマートフォンにつなぐと音楽アプリが起動して音楽が流れる）している^{||}。

一方で、喫煙中や、行列待ちでは、片手間にスマートフォンを操作している可能性があり、端末利用者の注目を得やすいと考えられる。本研究では、図 2.2 のように端末利用者がスマートフォン操作に集中しているかどうかによって、通知に最適なユーザ状況を定義する。

従来研究では、ユーザ状況推定のためにスマートフォンに搭載されている加速度やジャイロセンサを利用して、立つ、歩く、座る、走るなどの単純動作の認識や分析に関する研究が多く行われている [8, 9, 10, 11]。しかしながら、これらの手法では、着座や起立までは認識できても、例えば着座の中でゲーム操作をしているのか、喫煙しているのかまでは認識できない。すなわち、本研究で対象とする詳細な人の状況（例：喫煙中、乗車中、ゲーム中、工作中など）まで推定することは難しい。

通知タイミングを最適化するには、人の詳細な状況推定が必要となる。本研究では、スマートフォンで得られる情報から、これらの状況推定まで可能かどうか試行錯誤し、スマートフォンのタッチ操作状況（コンテキスト）を利用する事を着想した。タッチ操作コンテキストならば、様々な状況によりディスプレイへの挙動が変化すると考えられる。タッチ操作コンテキストの具体例として、スマートフォンの

[§]Tasker: <https://play.google.com/store/apps/details?id=net.dinglisch.android.taskerm>

[¶]SecureTask: <https://play.google.com/store/apps/details?id=com.balda.securetask>

^{||}xda-developers: <https://www.xda-developers.com/block-heads-up-notifications-gaming/videos/>



図 2.3: 通知タイミングとタッチ操作コンテキスト推定を実現するための必要技術

操作データから操作画面のどこがよく触られているかを解析することで、日頃操作する手や指(利き手)を推定できる可能性は高い。また、ユーザは喫煙中に煙草を持つために利き手ではない逆の手で操作している可能性がある。さらに、ゲーム中や仕事においては、集中した操作が想定されるため、他のアプリケーションよりもスワイプスピードやタッチ回数が多いなどの変化が考えられる。これらのタッチ操作の変化はスマートフォン上の様々なアプリケーション上で起こると想定される。

図 2.3 に示すように、このようなタッチ操作コンテキストをアプリケーションに依らず推定するには、シングル(マルチ)タッチ、シングル(マルチ)スワイプ、ピンチイン(アウト)、ローテートといった操作挙動や、各操作挙動に対する頻度や速度、圧力などを統計的に取得・分析し、タッチ操作コンテキストを、アプリ横断で推定する必要がある。しかし、Android に代表される OS(Operation System) は、Sandbox と呼ばれるセキュリティの観念から、アプリケーション同士でタッチ操作データのロギングが不可能であり、アプリケーション非依存なタッチ操作取得自体が非常に難しい。

以上を踏まえ、まず 2.2 節において、タッチ操作取得分析可能なシステム、TouchAnalyzer について詳述する。本システムは、本来不可能とされるアプリケー

ションに依らないタッチ操作の取得を，OS が管理するイベントデバイスファイルからタッチログデータを監視することで可能にする．しかし，OS が出力するログデータは，人間には可読しにくい複数行にわたって表された点情報の羅列（低水準のデータ）であり，何本指でのタッチやスワイプ操作挙動か，スワイプスピードや挙動頻度か，といった抽象度の高い情報（高水準のデータ）ではない．そのため，TouchAnalyzer は，低水準のデータを解析することで，8 本指までの操作を含むタッチ，スワイプ，ピンチイン（アウト），ローテートといったこうすいタッチ操作挙動を取得する．同時にスワイプ速度やタッチ操作頻度，接地面積，圧力といった統計情報を取得する．さらに，視覚的に解析しやすいよう，操作挙動をアプリケーションごとに色別に示したり，ヒートマップで表示する．

その後，2.3 節において，タッチ操作コンテキストの一つである操作形態（持ち方）を推定する手法について記述する．スマートフォンを支える手と操作する指の組み合わせを操作形態とし 12 種類を定義する．そのなかでも，本研究では，一般的とされる 8 種類の操作形態を対象に，機械学習アルゴリズムを用いて推定モデルを構築した．16 名の被験者データを利用して推定モデルを評価した結果，概ね 90% の確率で推定可能なことを確認した．

2.2 スマートフォンのタッチ操作情報取得手法

本節ではタッチ操作分析システムに関する関連研究について述べ，Android の一般的なタッチ操作取得経路の説明を行い，タッチ操作の情報をどこで取得するか検討する．その上でタッチ情報を取得する上での技術的課題について述べ，課題を解決する手法として OS のイベントデバイスファイルに出力される低水準のデータを解析し，タッチパネル上で行われるタッチ操作を取得，分析するシステム，TouchAnalyzer について記述する．最後に，開発システムの評価実験，およびその結果と考察について示す．

2.2.1 関連研究

タッチ操作コンテキストから人の詳細状況を推定するには、タッチ操作を取得分析し、タッチ操作コンテキストを推定する必要がある。

アプリケーション上でのタッチ操作を分析するサービスとしては、ClickTale Touch^{**}、Ptengine^{††}、Localytics^{‡‡}、USERDIVE for Apps^{§§}、Appsee^{¶¶}などが商用サービスを展開している。スマートフォンのアプリケーションや Web ページを解析対象として、どのボタンが押され、どのエリアがタッチされているのかを可視化、分析する機能を有する。これらの共通点として、専用の SDK を開発者に配布し、開発者はその SDK を組み込んだアプリケーションを作成する点である。タッチ操作は、SDK を通じて、専用のクラウドサービスに集約され、アプリケーション開発者は、Web サイトでその結果を閲覧可能である。SDK が提供されることで、アプリケーション開発者は簡単にタッチ操作分析システムを自身のアプリケーションに導入できる。

しかしながら、これらの商用システムで取得可能なタッチ操作は、開発者自身が SDK を組み込んだアプリケーションのみであり、他のアプリケーションも含め、スマートフォン上のタッチ操作をすべて取得できるものではない。また、たとえ SDK を組み込んでどのエリアがどの頻度でタッチされたかがわかるだけであり、タッチなのかスワイプなのか、1 本指なのか 2 本指なのかといった、タッチ操作挙動は一切取得できない。

また、多くの既存研究では、取得されたタッチ操作ログは、セキュリティやインタフェース改善に利用されている [12, 13, 14, 15, 16, 17, 18, 19, 20]。セキュリティ分野では、TouchLogger[12]、Touchalytics [13] が存在する。これらの研究は、過去に盛んに研究されていたキーボードの打鍵状況で個人認証を行う研究 [31, 32, 33] と同様に、ソフトウェアキーボードアプリケーションを使用した際のスワイプ状況や加速度の変化から個人認証を試みている。

インターフェース改善の研究として、黒澤ら [20] は、片手操作時のスワイプ方向

**Clicktale Touch: <http://www.clicktale.com/>

††Ptengine: <http://www.ptengine.jp/>

‡‡Localytics: <http://www.localytics.com/>

§§USERDIVE for Apps: <http://userdive.com/apps/>

¶¶Appsee: <https://www.appsee.com/>

に基づいた新しい操作手法を提案している．この研究では，ユーザのスイープ操作を観察し，右手親指で操作する人は，左上方向にスイープを行うことが少ないことを明らかにした．そして，その操作に対してスイープ以外の操作機能を割り当てている．操作を観察する手法は，OS が管理するイベントデバイスファイルからタッチログデータを監視する手法である．

しかしながら，これらの研究では，特定の操作スイープだけが対象であり，本研究のようにシングルタッチから 8 本指までのタッチ，2 本指を用いたピンチやローテートといったマルチタッチジェスチャを認識するアルゴリズム等については一切検討されていない．また，複数の機種や OS に関して調査されていない．

このように，従来研究において，取得可能なタッチ操作は，特定のアプリケーションを使用した時のみのものがほとんどである．アプリ横断で取得可能な黒澤ら [20] の研究においても，特定の機種や OS を対象にしており，複数の機種や OS を対象にしていない．さらに，取得されるタッチ操作が限定されている．

2.2.2 Android の一般的なタッチ操作取得経路

Android 上のタッチ操作は図 2.4 のような経路をたどって認識される．具体的には，以下のような流れで認識される．最初にユーザが画面を触ることでタッチパネルの静電容量が変化し，Touch Panel Driver がそれを認識する．次に Touch Panel Driver は認識した静電容量の変化をタッチログとしてイベントデバイスファイルと呼ばれる `/dev/input/eventX` (但し X は数字であり，これは端末ごとに異なる) に出力する．`eventX` に渡されたタッチログは，Application Framework(アプリケーションを実装する際に呼ばれるクラスやライブラリ) 内の System Server によって監視される．そして，System Server は，Application Process にタッチ操作の情報を渡す．

Android はこのタッチ操作取得経路の中で，タッチログ補完とリサンプリングと呼ばれる座標補正を行っている．タッチログ補完は，例えば Touch Panel Driver 部分で行うことで，過去のデータを用いて欠損値の補完を行ったり端末画面の端部分のデータを放棄したりしている．タッチログ補完がどこでどのように行われるかは，各端末メーカーの実装次第である．次に，座標補正は Application Process 内で

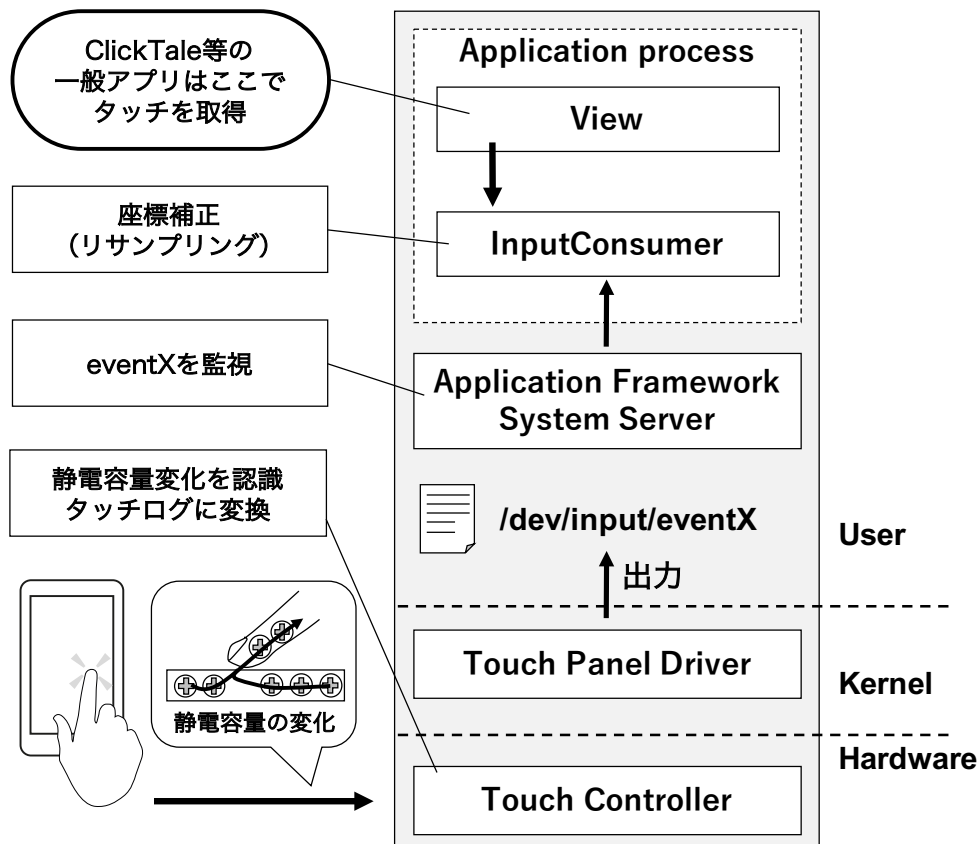


図 2.4: タッチ操作の認識の流れと提案システムとの関連

行われる．この座標補正はユーザが行うタッチ操作時の指の動きとアプリケーション画面の動きを一致させるため、つまり直感的操作でアプリケーション画面が滑らかに動くようにするために行われる．しかし、この座標補正されたタッチ操作の情報は、実際に端末操作者がタッチした場所のデータではなく、InputConsumerが独自にタッチ点を間引く、あるいは予測して出した値である．

2.2.3 タッチ情報取得方法の検討

上述した Android のタッチ操作取得経路を踏まえて、どの部分でタッチ操作の情報を取得するかを検討する．Android のタッチ操作の情報取得によく用いられる方法に図 2.4 の Application Process で SDK を利用する方法がある．SDK で組

み込む ClickTale などは，Application Process 部分でタッチ操作の情報を取得している．しかしながら，この部分で取得可能なタッチ操作の情報は，開発者が独自に SDK を組み込んだアプリケーションのみが対象である．つまり，本研究で目指す，他のアプリケーションも含めた全タッチ操作の横断的取得はできない．しかしながら，図 2.4 の `/dev/input/eventX` ならば，対象端末を USB 接続する事で全アプリケーション上のタッチ操作の情報をのぞき見できる．この `eventX` に出力されるタッチログの値と Application Process で取得されるタッチ座標値は座標補正の影響で異なるものの，タッチ操作の挙動を認識するという点では問題ない範囲の違いである．そこで今回は，あらゆるアプリケーションのタッチ操作を取得するため，`/dev/input/eventX` 部分でタッチ操作の情報を取得する方法を選択する．次項において，`/dev/input/eventX` 部分からタッチ操作の情報を取得する上での技術的課題について述べる．

2.2.4 タッチ操作を取得する上での技術的課題

タッチ操作を取得する上で解決すべき 2 つの課題が存在する．1 つ目の課題は，`eventX` (ここで `X` は数字) や `eventX` に出力されるログフォーマットが，各機種や Android のバージョンで異なることである．このことから，様々な機種を調査してその差異を明らかにする必要がある．2 つ目の問題は，`eventX` に出力される内容がある瞬間のタッチ点の情報であることである．スワイプなどの「操作」は，この点(一行の情報)を複数束ねた形式で表現されるため，各行を観ても，その行が 1 本指操作によって発生した点の 1 つなのか，2 本指操作か，あるいはタッチなのかスワイプなのかといったマルチジェスチャを含む具体的な挙動であると認識することは難しい．これらの各課題に対する解決方法については，以下の 2.2.5 項にて詳述する．次項では，Android 端末上の全タッチ操作ログをアプリケーションに関わらず横断的に取得し，その操作を分析(分類)，可視化するシステム TouchAnalyzer について説明する．

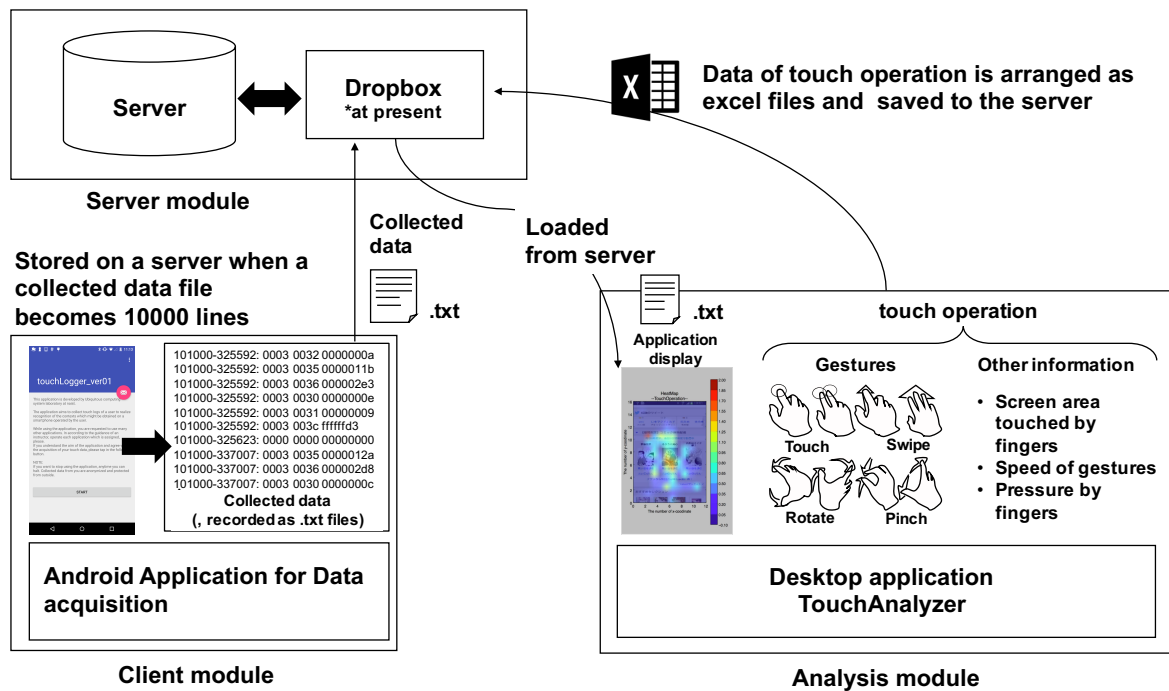


図 2.5: TouchAnalyzer の全体概要

2.2.5 TouchAnalyzer の設計と実装

本項では、提案システムの全体構成とシステムを構成する各モジュールの実装内容について詳述する。

TouchAnalyzer の全体概要

図 2.5 に示すように、提案システムは、1) クライアントモジュール、2) サーバモジュール、そして 3) 分析モジュールで構成されている。以下にそれぞれについて説明する。

クライアントモジュール

クライアントモジュールは、タッチイベントデータを記録し、サーバモジュールでアップロードする機能を持つ。クライアントモジュールは Android アプリケーションとして構築されており、root 権限を必要とする。また、クライアントモジュー

表 2.1: Android 端末および Android OS ごとにタッチイベントログのフォルダパスの違い

type of equipment	time format		number of eventX			Android
	- type	. type	1	2	6	
1 <i>GalaxyNexus</i>		o	o			4.2.1
2 <i>GalaxyS III</i>	o				o	4.0.2
3 <i>IdeapadTablet</i>	o				o	2.3.4
4 <i>XperiaArcHD</i>	o			o		4.0.4
5 <i>Nexus 4</i>		o	o			4.2.4
6 <i>GalaxyNote II</i>		o	o			4.1.2
7 <i>GalaxyS II</i>	o		o			4.0.3

ルはイベントデバイスログデータを貯めている，“\dev\input”を絶えず監視，記録し，対象イベントのログが 10,000 行になる毎にサーバーモジュールへ送られる．

この時，タッチイベントログのフォルダパスは，Android 端末開発メーカーや Android OS のバージョンによって完全に異なっている．例えば，Samsung Galaxy S III (Android OS 4.0.2) は，“\dev\input\event6”に出力されるが，Galaxy Note II (Android OS 4.1.2) では，“\dev\input\event2”にログが出力される．表 2.1 に，Android 端末および Android OS ごとにタッチイベントログのフォルダパスの違いをまとめたものを示す．将来的には，これらのパスを自動的に認識しタッチイベントデバイスログを取得する機能を実装することを考えている．

サーバーモジュール

サーバーモジュールはデータベースと API で構成される．現在は，対象のクライアント数が少ないため，一時的に Dropbox をサーバーモジュールとして利用している．

分析モジュール

分析モジュールでは，PC ローカル上で動作するシステム，TouchAnalyzer を開発している．このシステムは Python と matplotlib で構成される．TouchAnalyzer はサーバーモジュールからログデータをロードする．

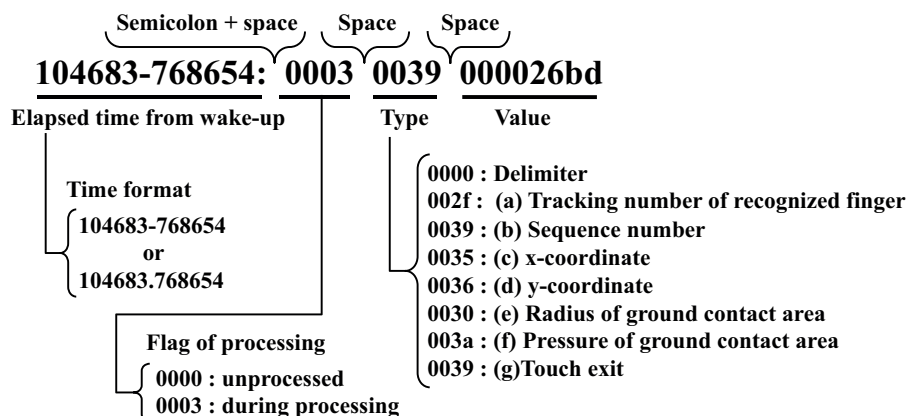


図 2.6: 出力されるログデータの 1 文

ここで、図 2.6 にロードされるログデータの 1 文を示す。出力データは、左から順に、起動経過時間、処理フラグ、種類、値を示している。起動経過時間とその他の出力は、セミコロンとスペースで区切られている。また、処理フラグ、種類、値の項目はスペースによって区切られ、これらの項目は全て 16 進数で表現されている。今回は、処理フラグ、種類、値の各項目を、説明の便宜上、1 列目、2 列目、3 列目として定義した。以下に、起動経過時間、処理フラグ、種類、値の項目内容について順に詳述する。

起動経過時間には 2 種類の時刻フォーマットが存在し、ハイフン「-」型、コロン「.」型がある。機種ごとの時刻フォーマットは、表 2.1 に示している。

1 列目の処理フラグには「0000」と「0003」2 つの値が出力され、各値はタッチイベントを処理していないか、あるいは処理中であるかを表している。

2 列目の種類は、実際に OS から出力されたログデータがどのような値を取得し、処理しているかを表している。処理フラグが「0000」の時は、必ず 2 列目に「0000」が出力される。そこで今回は、この「0000」の値を取得した情報の区切りとして「====」と表した。一方で処理フラグが「0003」の時、つまりタッチイベントが処理されている時は、(a) ~ (g) の処理内容が存在する。2 列目に「002f」が出力される場合は、(a) 認識した指の追跡番号を取得している。また「0039」が出力された場合には、(b) シーケンス番号と (g) タッチ終了が存在する。

表 2.2: 2 点タッチした際に出力されたログ

出力ログ例			ログ解析結果
列 1	列 2	列 3	タッチイベントに関する
1	0003	002f	0000001 (a) 認識した指の追跡番号
2	0003	0039	000001b1 (b) シーケンス番号
3	0003	0035	00000098 (c) x 座標
4	0003	0036	000001ca (d) y 座標
5	0003	0030	00000053 (e) 接地面積の半径
6	0000	0000	00000000 =====
7	0003	002f	0000000 (a) 認識した指の追跡番号
8	0003	0039	000001b2 (b) シーケンス番号
9	0003	0035	0000015c (c) x 座標
10	0003	0036	000001b1 (d) y 座標
11	0003	0030	0000005d (e) 接地面積の半径
13	0003	003a	00000006 (f) 接地部分の圧力
14	0000	0000	00000000 =====
15	0003	002f	0000001 (a) 認識した指の追跡番号
16	0003	0039	ffffff (g) タッチ終了
17	0000	0000	00000000 =====
18	0003	002f	0000000 (a) 認識した指の追跡番号
19	0003	0039	ffffff (g) タッチ終了
20	0000	0000	00000000 =====

この認識には 3 列目の値を用いている。もし 3 列目が「ffffff」ならば (g) タッチ終了を表すが、その他の値であれば、(b) シーケンス番号の取得を行っている。さらに、この 2 列目に「0035」が出力された場合はタッチした場所の (c) x 座標取得を表し、「0036」が出力された場合はタッチした場所の (d) y 座標取得を意味している。そして、「0030」が出力された場合は、タッチした部分の (e) 接地面積の半径取得を示している。この時、接地面積の単位は GalaxyS II の解像度が WVGA (Wide-VGA) の幅 800px、縦 480px であるため、 $[\text{rad} \cdot \text{px}^2]$ で表現される。2 列目で「003a」が取得された場合は、タッチした部分の圧力、つまり (f) 接地部分の圧力を示している。圧力単位は [hPa] である。

表 2.2 は、端末画面を 2 点タッチした際に出力されたログの例と、そのログを

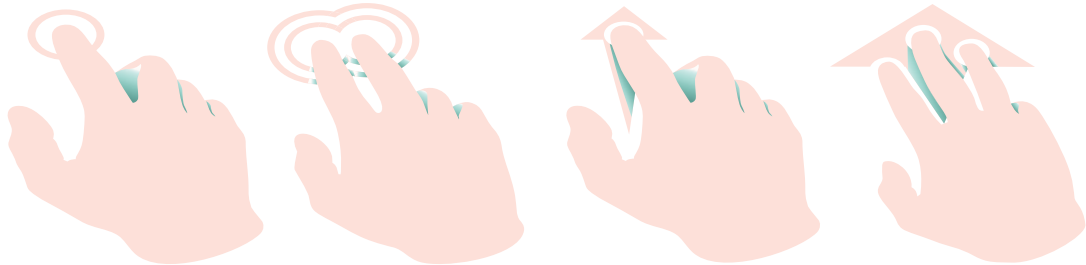


図 2.7: シングル・マルチタッチ

図 2.8: シングル・マルチスワイプ

図 2.6 の (a) ~ (g) を用いて解析した結果である．表 2.2 の 1 列目，2 列目，3 列目は，順に処理フラグ，種類，値の項目を示している．この出力からわかるように (a) ~ (g) は必ず出力されるわけではない．これは前回の値と差分がなかった場合に，event が値を出力しないためである．例えば今回のデータでは，(f) に関して出力していない場合がある．これは接地部分の圧力が，各指で異なったためであると考えられる．また，(a) はマルチタッチ時のみ出力される．

2.2.6 タッチ操作の認識

タッチ操作を分析するため，TouchAnalyzer を用いてタッチ操作の挙動を認識する．認識する挙動は図 2.7 のシングルタッチ・マルチタッチ，図 2.8 のシングルスワイプ・マルチスワイプ，図 2.9 のピンチイン・ピンチアウト，図 2.10 のローテートである．これらの挙動は一般的に多くのユーザがスマートフォン上で頻繁に行うタッチ操作の挙動であり，タッチ操作のほとんどを占めている．もしこれらの挙動の認識とユーザが使用しているアプリケーションのデータが同時に取得できれば，アプリケーションごとにどの挙動や操作が頻繁に行われているか，といった情報が取得可能となり，同じアプリケーションでも端末使用者によって UI が変化する

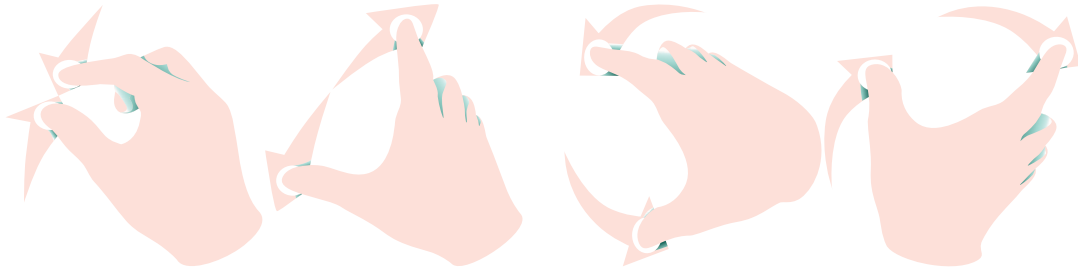


図 2.9: ピンチ

図 2.10: ローテート

る、今までにないダイナミック UI が実現可能になると考える。さらに、これらの情報はタッチ操作の挙動に対する様々なコンテキストを推定する上で必要になると考える。

これを踏まえた上で、今回は事前に各タッチ操作に対してどのようなタッチログが出力されるかを観察した。この観察結果を利用して、タッチ操作ごとの認識アルゴリズムを提案し、詳述する。

シングルタッチ・マルチタッチ

端末上で行われたタッチ操作がシングルタッチかマルチタッチかを判断するため、表 2.2 に示した (a) 認識した指の追跡番号が使われている。もし (a) が取得されなければ、シングルタッチであると判断される。(a) が取得された場合は、認識した指の数だけ (b) ~ (f) の値を取得し、認識した各指に対して追跡が行われる。タッチイベント終了の際も同様に行われ、(a) で認識した指の追跡番号の分だけタッチイベントの終了処理が行われる。ここで、タッチイベント処理の流れを Algorithm 1 として示す。例えば、端末画面を 3 本の指でタッチした場合は、画面に触れているそれぞれの指に追跡番号が割り振られる。そして、指ごとに (b) ~ (f) の値を取得、追跡していく。ここで、もし仮に 3 本の指で画面をタッチした後、いずれかの

Algorithm 1 シングルタッチ・マルチタッチ時の処理

```
if (a) が取得されない then
  表 2.2 の (b) ~ (f) を取得
else {(a) が取得される}
  for (a) 認識した指の追跡番号がある && 追跡番号が 10 より小さい do
    認識した指の追跡番号ごとに表 2.2 の (b) ~ (f) を取得
  end for
end if
```

指を 1 本画面から離して残り 2 本で画面を操作する場合は、追跡番号が割り振られている指のみを認識し続け、画面から離された指に関しては、タッチ終了の処理を行う。端末画面上で認識される指の数は最大で 10 本であることを確認できた。

シングルスワイプ・マルチスワイプ

スワイプはシングルタッチ・マルチタッチが連続で行われていると認識される。つまり、追跡している指ごとに終了判定が行われる（指が離れる）まで連続で Algorithm 1 を繰り返してタッチログを取得し続ける。これにより、シングルスワイプ、マルチスワイプを認識している。

ピンチ

ピンチにはピンチイン・ピンチアウトが存在する。これらは 2 点マルチスワイプ時に起きる特別な挙動である。追跡対象の 2 つの指が端末画面に接着し、離れるまでタッチログを取得し続け、始点と終点、そしてその軌跡を取得する。そして取得した 2 つの終点 P_3, P_4 の間の距離を $L1$ 、始点 P_1, P_2 の間の距離を $L2$ として求め、これら $L1, L2$ の大小比較からピンチイン・ピンチアウトを認識している。図 2.11 の左にピンチの始点・終点の関係を示す。

ローテート

ローテートは 2 点タッチと 2 点スワイプを組み合わせた特別な挙動である。スワイプ動作の中でも、タッチログの軌跡が円を描くような場合に認識される。追跡対

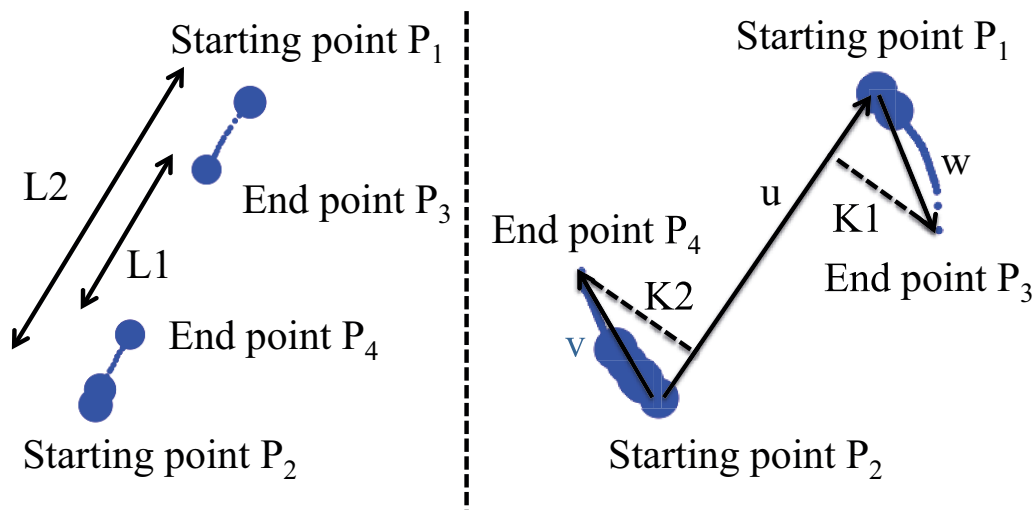


図 2.11: ピンチとローテートの定義

象の指が端末画面に接着し、離れるまでタッチログを取得し続け、始点、終点とその軌跡を取得する。この時、終点 P_3 と始点 P_1, P_2 を結ぶ直線へ垂線を引いた距離を $K1$ 、終点 P_4 から始点 P_1, P_2 を結ぶ直線へ垂線を引いた距離を $K2$ とし、 $K1$ と $K2$ の長さがある閾値以上であれば、ローテートであると認識する。今回は始点 P_1, P_2 を結ぶベクトルを u 、始点 P_1 と終点 P_3 を結ぶベクトルを v 、始点 P_2 と終点 P_4 を結ぶベクトルを w とし、外積計算から $K1$ と $K2$ を求めてローテートを認識する。図 2.11 の右にローテートの始点・終点の関係を示す。

2.2.7 タッチ操作の統計分析

認識したそれぞれの挙動ごとに頻度と平均速度を出力する。ここでの頻度は、タッチ操作の挙動を認識した回数として定義する。次に平均速度に関して詳述する。ある点 $P(x_p, y_p)$ から点 $Q(x_q, y_q)$ までの距離を、時間 T_p (秒) から T_q (秒) で移動した時の速度 v_{pq} を求める場合、一般的に以下の式 (2.21) が適用される。

$$v_{pq} = \frac{\sqrt{(x_q - x_p)^2 + (y_q - y_p)^2}}{T_q - T_p} \quad (2.21)$$

しかし、タッチ操作の挙動においては、図 2.12 に示すような軌跡がたびたび (特

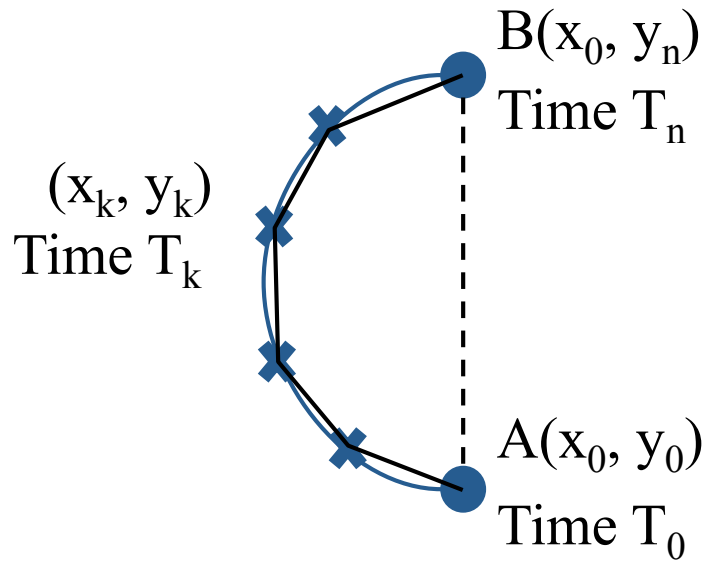


図 2.12: タッチ操作の挙動に見られる軌跡

にスワイプ時) が見られる。図 2.12 は，端末利用者がタッチパネル上のある点 A (x_0, y_0) から点 B (x_0, y_n) までを， T_0 (秒) から T_n (秒) の間にスワイプした軌跡を示している。スワイプはタッチの連続として認識されるため，点 A から点 B までの間に，時間 k (秒) における点 (x_k, y_k) が n 個存在する。この場合，点 A と点 B の x 座標は等しいため，式 (2.21) では適切な速度を取得できない。本研究では，挙動ごとの平均速度を出力する計算式を考案し，TouchAnalyzer に実装する。タッチ操作挙動として取得した描画点が n 個あった時，求める速度を式 (2.22) のように定義する。

$$v_{ab} = \frac{\sum_{k=0}^{n-1} \frac{\sqrt{(x_{k+1} - x_k)^2 + (y_{k+1} - y_k)^2}}{T_{k+1} - T_k}}{n - 1} \quad (2.22)$$

式 (2.22) は，各点ごとの速度を取得して足した後，全体の速度の平均をその挙動の速度としている。

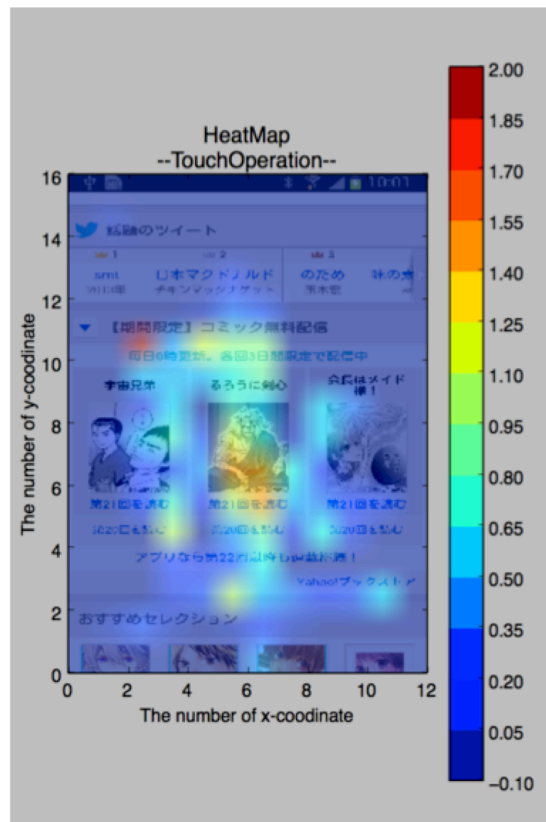


図 2.13: ヒートマップによるタッチ操作の可視化

2.2.8 タッチ操作の可視化

解析したタッチ操作のログデータを用いて、タッチ操作の可視化を行う。Touch-Analyzer は取得、解析したタッチ操作挙動の x, y 座標を点として描画する。この時、取得データにタッチ面積の半径が含まれる場合は、タッチ面積の半径に合わせて描画する点の半径が変化するように設計する。また、開発システムは foreground にあるアプリケーションのタッチ操作の挙動を常に取得する。この時、タッチ操作の挙動の取得はアプリケーションに非依存で行われ、foreground アプリケーションがユーザのタッチ操作により切り替わるたび、描画する色は変化する。さらに、図 2.13 に示すように、開発システムはヒートマップによるタッチ操作の表示が可能である。このヒートマップ表示では、スクリーンの各部分について、触れた頻度が高

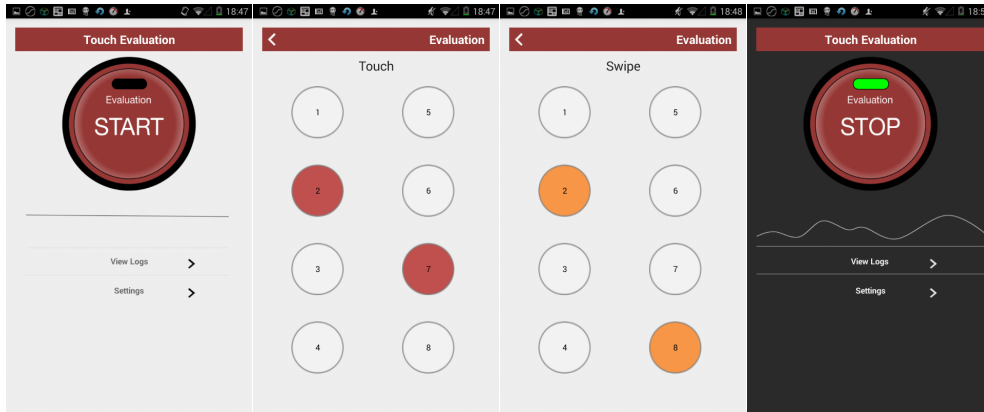


図 2.14: アプリケーションの実験画面

いほど赤く，頻度が低いほど青く表示する．また，ヒートマップの背景部分には，ユーザが操作しているアプリケーションの画面キャプチャをリアルタイムに出力している．これにより，どのようにユーザがアプリケーションの画面を操作したか，どの機能を操作しているかを視覚化している．

2.2.9 TouchAnalyzer 評価実験および評価方法

本研究で開発したシステム TouchAnalyzer に関する評価実験を行う．コンテキスト推定に向けたタッチ操作のセンサ利用には，アプリケーション上で認識されるタッチ操作の挙動を正確に認識できていることが重要である．そこで，タッチ操作の挙動認識に関する評価を行う．

TouchAnalyzer を評価するにあたり，本研究では開発した評価用のシステム（アプリケーション）を利用する．この評価用のシステムを GalaxyS2，OS バージョン 4.0.3 にインストールして実験を行う．本実験では 1 名の被験者が，評価用のアプリケーションが出力する 1～8 までの番号が割り振られたボタンをタッチする，あるいはそのボタンを始点にスワイプを行う．タッチが認識された場合は，ボタンの色が赤くなり，スワイプが認識された場合は，ボタンがオレンジに変化する．これをシステムが対象とするタッチ，スワイプのすべてに対して行う．図 2.14 にアプリケーションの画面を示す．なお，ピンチ，ローテートに関しては，評価用のアプ

表 2.3: 評価用アプリケーション上でのタッチ操作挙動の認識精度

タッチ操作の挙動タイプ	タッチ操作挙動の認識精度 (正解率 $x\%$)							
	タッチ操作で使用した指の本数							
	1	2	3	4	5	6	7	8
touch	100.0	98.0	100.0	94.0	97.0	99.0	100.0	99.0
swipe	100.0	100.0	100.0	95.0	97.0	97.0	86.0	94.0
pinch-in	-	98.0	-	-	-	-	-	-
pinch-out	-	100.0	-	-	-	-	-	-
rotate	-	85.0	-	-	-	-	-	-

リケーションは使わず，これらの挙動が最も行われやすい，Google Maps を用いて評価する．この時，ピンチイン時に地図が拡大する，ピンチアウト時に地図が縮小する，そしてローテート時に画面が回転した場合を，正解値とする．式 (2.23) に評価メトリックとして用いる正解率を定義する．

$$x = \frac{a}{b} \times 100 \quad (2.23)$$

開発システム TouchAnalyzer での認識回数を a (回)，評価アプリケーション上での認識回数を b (回) とした時，正解率 $x(\%)$ としている．ここで，評価アプリケーション上での認識回数は 100 回として実験を行う．

結果および考察

表 2.3 に実験結果を示す．今回評価した全てのタッチ操作の挙動認識精度は，85% 以上を示す．ここで，誤認識になった原因について考えられる点を述べる．最も可能性の高い原因として，OS がタッチログを出力しなかったことがあげられる．TouchAnalyzer は，OS のログデータを解析している．このとき TouchAnalyzer は，本来携帯端末ドライバ (図 2.4 の Touch Panel Driver) が行うような座標補完を行っていない．そのため，OS がログデータを出力しない場合は，タッチ操作の挙動を認識できない．また，TouchAnalyzer はタッチログに x, y 座標の両方が含まれる時のみをタッチ操作の情報として認識し，タッチ操作の挙動認識データに用いている．つまり，システムは OS がタッチログを確実に出力した時のデータを用いて，ジェスチャの認識を行っている．実際に，2 本指タッチ操作で 2 回の誤認識があり，この時システムはその 2 回を 1 本スワイプと認識している．具体的に

表 2.4: 指の違いによるシングルスワイプの認識精度の違い

指の違いごとのシングルスワイプの認識精度 (正解率 $x\%$)	
タッチ操作の挙動時の指の種類	タッチ操作で使用した指の本数
	1
親指	100.0
人差し指	100.0
中指	100.0
薬指	90.0
小指	76.0

は、OS が出力したデータは 2 本の指の内の 1 本は x, y 座標を適切に出力していなかった。これは全挙動に対して起こりうる。

スワイプになると、その影響はより大きくなると考えられる。OS が認識できない原因は、タッチパネルの精度 (タッチパネルをタッチした際の指先の大きさ、つまり指とタッチパネルの設置面積の大きさを認識できるか) に依存すると考えられる。実際に、小指を用いて行う 7 本スワイプの認識精度は、86% と他の認識精度より低くなっている。これを確認するため、指が異なる際のシングルスワイプの認識精度を式 (2.23) を使用して評価する (表 2.4)。このとき、 $b = 100$ である。指とタッチパネルの接地面積が小さくなりやすいと考えられる小指の認識精度が最も低い。これは子供を対象として認識する場合にも影響すると考えられる (子供は指が小さいため)。

また、認識精度が他より低いものにローテートがある。ローテート以外と認識した誤認識の 15 回は、ピンチイン、ピンチアウトもローテートと同時認識している。ピンチと認識しつつ、ローテートであると認識した場合を含めると、はローテートを 100% 認識している。一般的に、ローテートを行いつつピンチを行うことは頻繁に見受けられる。これより、この誤認識は許容できるものとする。

本手法の有効性に関して考察する。本手法は、将来的に端末メーカーが採用し、root 権限が必要な場所で利用されることを想定して開発している。Android 端末メーカーのみが触れている (OS の一部として利用される) 部分を利用しているため、他アプリケーションがタッチログデータを見ることはできない。また、タッチ、スワイプ、ローテート、ピンチなどの挙動や速度などの情報は、多くの研究や実用



図 2.15: 人の詳細状況により変化する持ち方（操作形態）の例

システムに利用されているため、タッチ操作の挙動情報は、プライバシー情報ではないと考える。以上より、本手法は Android 端末のネイティブなシステムとして有用であると考えられる。

2.3 タッチ操作分析に基づく操作形態推定手法

タッチ操作コンテキストから人の詳細状況を推定するには、どのような事を想定すれば良いか検討する必要がある。本研究で想定する人の詳細状況とは、喫煙中や、行列待ち、ゲーム中など、既存研究では判別できなかった細かな状態を指す。このような詳細状況においてタッチ操作コンテキストは、図 2.15 のように変化することが考えられる。例えば、喫煙中の場合、煙草を持ちながらスマートフォン操作を行なうため、普段と端末の持ち方が変化すると想定される。ゲームなどにおいても同様の現象が起きる可能性は高い。このように、いつもと違う端末の持ち方（以下、操作形態）を推定できれば、今まで判別できなかった人の詳細な状況までを推定できる可能性がある。

本節では、タッチ操作コンテキストの一つである操作形態を、タッチ操作ログをセンサとして利用し、推定する手法について述べる。最初に操作形態の定義について示し、関連研究から導かれるシステム要件を記載した上で、操作形態推定手法を提案し、提案手法の評価と考察を行う。

2.3.1 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 [21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33]。本項では、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。

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 [21] 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 [22] 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依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。例えば、Harison ら [23] や Kim ら [24] は、静電タッチセンサを対象デバイスに組み込む事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 [25] 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 [26] 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。

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 [27, 28] 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 ContextType[29] 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．これらの手法の目的は文字の誤入力を改善することが目的である．そのため，本研究が対象とする持ち方とは異なる．また，ContextType は複数のセンサデータを利用するため複雑なデータ処理が必要となる．

本研究の狙い

2.3.1 項で述べた従来の推定手法には，解決すべき以下の 2 つの問題点が存在する．

- (1) 特殊な追加センサを使用するため汎用性が保てない点
- (2) 複数センサのデータ処理が必要なため端末の負担が大きい点

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，かつ端末への負荷が少ないスマートフォンの操作形態推定手法を提案するという点で既存研究と異なっている．

2.3.2 操作形態の定義

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 要素の組み合わせ（以下，操作形態）とし，3 要素が取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，その他の指（主に人差し指）}

ここで，支える手がない場合とは，スマートフォンが机の上などに置かれている状況のことを指す．操作する指に関して，今回はユーザの指の使用頻度を考慮して親指とその他の指（主に人差し指を想定しているが，人によって中指などを使用する







	支持		操作		
	手	手	手	指	
①	右	右	親	→	
2			その他		
3	左	右	親	→	
④			その他		
⑤	無し (机上)	右	親	→	
⑥			その他		
⑦	左	左	親	→	
8			その他		
9	右	左	親	→	
⑩			その他		
⑪	無し (机上)	左	親	→	
⑫			その他		

図 2.16: 操作形態の種類

人もいる)とした。

定義に基づいて組み合わせを考えると 12 種類の操作形態が考えられるが、右手で支え、右手の人差し指等で操作するという形態は一般的とは言い難い。また、左手で支え、右手で操作する場合、人差し指等で操作するのが一般的と考える。中には、親指等で操作するユーザが存在することは確認されているが、本研究ではその場合でも人差し指で操作したこととして分析する。実際に、持ち方に関するアンケート調査***では、スマートフォンを片手で支え、その親指で操作している人が 67.7% である。両手持ちの親指操作が 12.0%、両手持ち人差し指操作が 16.1% となっており、これらの持ち方は全体の 95.9% を占めている。以上より、今回の操作認識では、ほぼすべての持ち方をカバーしていると考えられる。

結果として、図 2.16 に示すとおり、端末操作者は 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 2.16 では例として、支持手がある

*** <https://www.orefolder.net/blog/2017/07/hands-enquete-result-2017/>

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。

2.3.3 要件設定

本研究では、特殊なセンサを利用せずに、アプリケーションを横断して端末に負担をかけず 8 種類の操作形態を認識することを目的とし、スマートフォンの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。タッチ操作情報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。本研究では、アプリケーションに依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言いがたい。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。

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 のイベントデバイスログデータは、複数行に渡って表現される点情報の塊（低水準のデータ）であり、人間には一見しても理解しがたい。さらに、端末ごとにイベントデバイスログのデータフォーマットは異なる。そのため、イベントデバイスログ

のような低水準のデータから操作形態の認識を実現するには、操作形態時のタッチ操作情報を、統計的に分析しやすい形へ変換する必要がある。例えば、ユーザがスワイプ動作時に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、操作範囲などの抽象度の高い情報（高水準のデータ）に変換し、特徴などを分析する必要がある。本研究では、要件 1 に対応するため、開発システム TouchAnalyzer を活用する。

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。例えば、操作形態を認識するため、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、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。推定プロセスについては、2.3.4 項で述べる。

2.3.4 推定プロセスの全体概要

図 2.17 に推定プロセスの概要を示す。今回、推定したい操作形態は 8 種類である。この 8 種類の操作形態は、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、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。実際に、Berchtold らは 20 名から加速度センサのデータを収集、利用して、機械学習アルゴリズムを用いることで、机の上の状態を 90% 以上で認識している [34]。本研究では、これらの手法が既に広く普及していると仮定し、加速度を用い

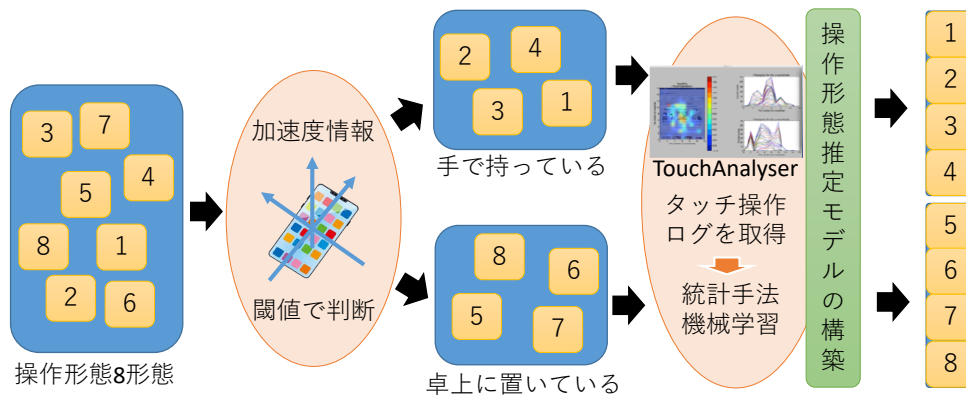


図 2.17: 推定プロセス

て 8 種類の操作形態を 4 種類ずつに分ける．次に，それぞれの 4 種類の操作形態を TouchAnalyser から得られるタッチ操作ログ等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ものが望ましい．具体的な特徴量および数理モデルについては，2.3.5 項で述べる．

2.3.5 操作形態推定手法の実現

本項では，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のに関して詳述する．

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) に協力を依頼する。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、実験端末には端末サイズがそれぞれ 5 インチ, 6 インチである, Nexus 5 および Nexus 6 を各 5 台ずつ利用する。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事で被験者のログデータを収集する。

実験参加者は図 2.16 に示した 8 種類の操作形態に対し 3 分間ずつ, 計 1 時間 30 分の操作を行う。実験中に操作するアプリケーションは, Yahoo!ニュース^{†††}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, 特徴量として利用するため参加者の手の大きさを計測する。

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 名のデータを利用する。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, データ収集が出来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。

特徴量

本研究では, 操作形態推定モデルを構築するため, 2.3.3 項の要件 2 で示した, ユーザや端末の不可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する。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。例えば, 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, 操作形態に関連す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 }

^{†††}Yahoo!ニュース: <https://news.yahoo.co.jp/promo/app/yjnews/>

- 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時、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。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する。

列挙した 9 つの特徴量をスワイプが行われる度に抜き出し、全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)を作成する。

一方で、データセット 1 にスマホ以外から得られる情報を加えた場合、どのような結果になるかを確認するため、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た、データセット 2(以降、データ 2)を作成する。

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 交差検証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。

アルゴリズムの検討

2.3.3 項の要件 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なう必要がある。今回は比較的シンプルなモデルとして決定木 (Decision tree) とロジスティック回帰 (Logistic regression) を選択した。また高精度で推定できるといわれている代表的なアルゴリズムとして、ランダムフォレスト (Random Forests) 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。

1. 決定木 (Decision tree)

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

ムである C4.5 決定木を用いた。

2. ロジスティック回帰 (Logistic regression)

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 (Random Forests)

ランダムフォレスト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、判定速度も高いと言われているが、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。

2.3.6 結果

2.3.5 項で述べたデータ 1(支持手の指の長さ情報が、無いデータセット) とデータ 2(有るデータセット) に対して、機械学習アルゴリズムの決定木 (Decision tree)、ロジスティック回帰 (Logistic regression)、ランダムフォレスト (Random Forests) で推定モデルを構築し評価した結果を詳述する。

総合的な評価結果を表 2.8 に示す。Nexus 5 および Nexus 6、どちらも同様の傾向が見られたため、2.3.6 項では、特に Nexus 5 についての評価結果を記載する。

また、8 種類の操作形態について、Nexus 5 および Nexus 6 それぞれに対し上記のアルゴリズムから構築した推定モデルによる推定結果は、本項の「総合推定結果」部分で述べる。

表 2.5: データ 1 に対する nexus 5 での評価結果

(a) 決定木 (Decision Tree)

Class	Precision	Recall	F 値
1	0.602	0.606	0.581
2	0.594	0.598	0.584
5	0.608	0.575	0.545
6	0.432	0.434	0.429
Average	0.562	0.563	0.543

(b) ロジスティック回帰 (Logistic regression)

Class	Precision	Recall	F 値
1	0.697	0.678	0.635
2	0.746	0.713	0.601
5	0.670	0.678	0.650
6	0.518	0.499	0.438
Average	0.660	0.644	0.612

(c) ランダムフォレスト (Random Forests)

Class	Precision	Recall	F 値
1	0.683	0.692	0.678
2	0.636	0.631	0.614
5	0.682	0.627	0.586
6	0.467	0.463	0.444
Average	0.602	0.596	0.576

表 2.6: データ 2 に対する nexus 5 での評価結果

(a) 決定木 (Decision Tree)

Class	Precision	Recall	F 値
1	0.979	0.979	0.979
2	0.977	0.976	0.976
5	0.991	0.988	0.989
6	0.690	0.736	0.670
Average	0.898	0.923	0.899

(b) ロジスティック回帰 (Logistic regression)

Class	Precision	Recall	F 値
1	0.845	0.768	0.707
2	0.954	0.956	0.954
5	0.988	0.980	0.984
6	0.863	0.775	0.759
Average	0.884	0.853	0.822

(c) ランダムフォレスト (Random Forests)

Class	Precision	Recall	F 値
1	0.989	0.989	0.989
2	0.971	0.972	0.971
5	0.995	0.997	0.996
6	0.928	0.887	0.889
Average	0.967	0.969	0.965

Nexus 5 における操作形態推定結果

表 2.5 および表 2.6 に、データ 1、およびデータ 2 を用いた時、実験端末である Nexus 5 に関して評価した結果を示す。各表のクラス番号 1, 2, 5, 6 は、図 2.16 での 1, 4, 7, 10, にそれぞれ対応しており、各クラスの Precision, Recall, F 値を記載している。

加えて評価に leave one person out 交差検証を採用している。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 2.5 の (a) 決定木 (Decision Tree), (b) ロジスティック回帰 (Logistic regression), (c) ランダムフォレスト (Random Forests), それぞれに関して平均 Precision, Recall, F 値をみると、最も良かったのはロジスティック回帰アルゴリズムの推定モデルであり、Precision が 66.0%, Recall が 64.4%, F 値が 61.2% である。

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ることがわかる。ロジスティック回帰アルゴリズム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 2.7(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。

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 2.6 の (a) 決定木 (Decision tree), (c) ロジスティック回帰 (Logistic regression), (b) ランダムフォレスト (Random Forests) 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ランダムフォレストアルゴリズムの推定モデルで、平均の Precision, Recall, F 値はそれぞれ、96.7%, 96.9%, 96.5% である。またロジスティック回帰アルゴリズム、決定木の順で推定精度が下がることがわかる。各平均精度の値はロジスティック回帰アルゴリズムでの推定モデルで、88.4%, 85.3%, 82.2%, 決定木での推定モデルでは、89.8%, 92.3%, 89.9% である。

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することがわかる。最も推定精度が良かったランダムフォレストアルゴリズム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 2.7(b)) に示す。この被験者は、ほぼ 100% の精度で推定可能である。このため、

表 2.7: 実験参加者 1 名の場合の混同行列

(a) ロジスティック回帰

	1	2	5	6
1	14	23	0	1
2	21	27	0	0
5	0	0	42	20
6	0	0	20	29

(b) ランダムフォレスト

	1	2	5	6
1	75	0	0	0
2	0	54	0	0
5	8	0	138	0
6	0	0	0	80

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わかる。

総合推定結果

Nexus 5, Nexus 6 における操作形態推定結果に対する平均の F 値を表 2.8 に示す。表のクラス番号 1, 2, 5, 6 は上記結果と同様に対応している。一方で、表クラス番号 3, 4, 7, 8 は図 2.16 での 5, 6, 11, 12 に合致している。

推定結果について Nexus 5, Nexus 6 に対し、クラス 1, 2, 5, 6 の推定精度を確認する。まず Nexus 5 に関してみると、データ 1 の時はロジスティック回帰アルゴリズムの場合が最も高く、61.2% である。データ 2 に対してはランダムフォレストが最も高く、96.5% であった。一方でクラス 3, 4, 7, 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共にランダムフォレストであり、53.7%, 95.2% である。

Nexus 6 に関して表 2.8 を確認する。クラス 1, 2, 5, 6 の推定精度をみると、データ 1, データ 2 に対する最高精度を出したのはどちらもロジスティック回帰で構築した推定モデルである。各精度は 48.9%, 89.9% となる。クラス 3, 4, 7, 8 に関しては RF がデータ 1, データ 2 に対して最も良い推定精度である。その時の F 値は、それぞれ 52.9%, 85.0% であることがわかる。

表 2.8: 総合評価：すべての組み合わせの F 値一覧

機種 手法		Nexus 5			Nexus 6		
		決定木	ロジスティック回帰	ランダムフォレスト	決定木	ロジスティック回帰	ランダムフォレスト
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	0.965	0.844	0.899	0.842
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	0.952	0.847	0.838	0.850

2.3.7 考察

本項では、推定結果に基づいて、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。

提案手法の有効性

2.3.5 項では、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、ロジスティック回帰、ランダムフォレスト）を用いて比較した結果、概ね 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、さまざま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にも考えられる。例えば、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。

一方で、今回の機械学習アルゴリズムを用いた持ち方認識手法では、特徴量に採用した指の長さが有効である。今後システムの有効性をさらに高めるには、指の長さを計測せずに操作形態を推定することが望ましい。これを実現するには、身長や性別などを簡単なアンケートや声から指の長さを推定し、個人で操作形態を選択できるよう実装する必要がある。例えば、手の大きさと身長、性別に関して、AIST では日本人の手の大きさと様々な身体情報（身長、性別、体重など）を計測し、公

開している^{†††}AISTの公開データを利用すれば、これらの情報と、本研究で計測した被験者の手の大きさを比較することで、指の長さを計算できる。計算した指の長さを、推定モデル構築に利用したデータセットに組み込むことで、操作形態を今回と同程度に推定できる可能性は高い。そこで、将来的に、提案した操作形態推定モデルを組み込んだアプリケーションを配布する際に、身長や性別を入力してもらうことで、操作形態推定を実現できる。

また一方で、身長と声の高さ(基本周波数)には弱い相関が有ることが知られている^{§§§}[48]。そのため、上記のように身長と性別の入力項目を設けなくとも、ユーザの声を任意に取得できるようにすることで、入力フォームを削除でき、さらにコンテキストウェアネスを高められる。これらの情報が実際に操作形態推定に利用可能か、推定精度がどうなるかを実装を含め確認することが、今後の残された課題である。

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、ランダムフォレストが約95%と他の手法と比べて高い精度を示す。しかしながら、Nexus 6では、どの手法もおおよそ85%となり、モデル間の差はほとんど見られない。精度面だけを考えると、ランダムフォレスト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、ランダムフォレストに次いで良い結果となる。また、計算量はランダムフォレストよりも少ないことは明らかである。2.3.3項で述べた要件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言えない。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、同サイズそ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。

^{†††}AIST 日本人の手の寸法データ: <https://www.dh.aist.go.jp/database/hand/index.html>

^{§§§}MRIにより計測した声道長と音声データとの関係: http://www.konan-u.ac.jp/hp/kitlab/pub/asj201203_hatano_slide.pdf

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。

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みなし、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。本研究では、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う。1章で述べたように、操作形態がわかることによって、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言える。例えば、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る。例えば、ATM搭載のタッチパネルを強く押す、かつ早く押すユーザには入金ミスをしていないか確認するように通知をだすことが考えられる。また対話型ロボットとのコミュニケーションでタッチパネルを利用する場合は、利用者の操作の癖や押し方の分析によって、ユーザが操作に慣れていないかや、急いでいるか、イライラしていないかなどを検知し、UIの変更や、利用および生活へのアドバイスなどを行うサービスが実現できる。これらを踏まえ、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。

2.4 まとめ

本章では、タッチ操作に基づくコンテキスト推定に必要な基盤として、Android スマートフォン上で発生する全タッチ操作を取得、記録、可視化するシステムの提案と実装を行った。タッチ操作の挙動を 100% 認識できることを目指し、評価用アプリを用いて提案システムの基本的な認識率を評価した結果、使用頻度の高い 1 本指タッチやスワイプ、2 本スワイプは 100%、2 本指タッチは 98% の精度で認識できた。

また、TouchAnalyzer を利用し、ユーザによるスマートフォンの操作形態（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。マルチ操作を含むタッチやスワイプなどのタッチ操作挙動が発生したスクリーン上の場所、および頻度を特徴とした操作形態を推定するモデルを、様々な機械学習法により構築し評価した。同じ操作形態であっても個人差のため、精度が悪くなるという課題を、操作する指の長さを特徴量として追加することで、解決をはかった。最終的に、16 名の実験参加者による 8 種類の操作形態をランダムフォレストアルゴリズムを用いることにより、F 値約 90% で推定できた。残された課題は、操作形態の推定結果を用いたアプリケーションの提供を通じて、本手法の有効性を評価して検討することである。

第 3 章 BLE を利用したグループ検出手法

第 2 章では，タッチ操作解析による人の状態（コンテキスト）推定について述べてきたが，適切な情報通知タイミングに基づくインテリジェント通知の実現には，同伴者コンテキストの推定も非常に重要である．同伴者コンテキストがわかれば，同伴者が会社関係（上司か同僚）の人なのか，もしくは友人なのか，それとも初対面の人なのか等を推定でき，通知するかどうかの判定を行うことのできる，インテリジェントな情報通知タイミングの制御が実現できる．本章では，まずはじめに，適切な情報通知タイミングと同伴者コンテキストの関連について説明する．それを踏まえ，同伴者コンテキスト推定に向けた必要技術として，BLE（Bluetooth Low Energy）を用いたグループ検出手法について詳述する．

3.1 あらまし

従来通知では，適切なユーザ状況（通知に良いタイミング）で，誰と居るか（同伴者コンテキスト）を考慮していない．そのため，適切なタイミングでの通知が行われていない現状がある．例えば，職場において，タッチ操作コンテキストからユーザ状況が喫煙中と認識できた時，同伴者が上司だった場合は上司に敬意を払う必要があるため，端末操作を促すような通知は控える必要がある．一方，同伴者が同僚であった場合は，特別な場合や状況を除いて，通知することに大きな問題はないと想定される（図 3.1）．このように同伴者によって通知するかどうか判定できるような賢さを持つ情報タイミングの制御は，同伴者コンテキストを正しく推定できることで実現できる．

同伴者コンテキストを推定するには，対象者が誰と居るのかを，どのような手法を用いることで推定できるか検討する必要がある．図 3.2 に，同伴者コンテキスト推定に関する必要技術について示す．具体的には，同伴者が会社関係（上司か同僚）の人の場合，対象者と同伴者は日常的に行動していると考えられる．つまり，ユーザと同伴者の間の物理的な距離が近い（つまり，近接性が高い）可能性が高い．一方で，同伴者が日常とは異なるグループで行動している場合，同伴者は友人，もしくは初対面の人と推定できる可能性がある．つまり，同伴者コンテキストを推定す

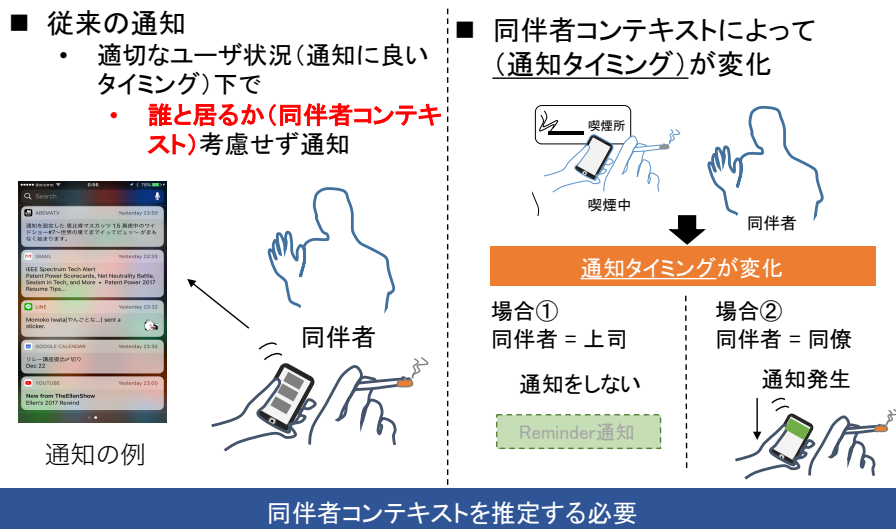


図 3.1: 適切な情報通知タイミングと同伴者コンテキストの関連

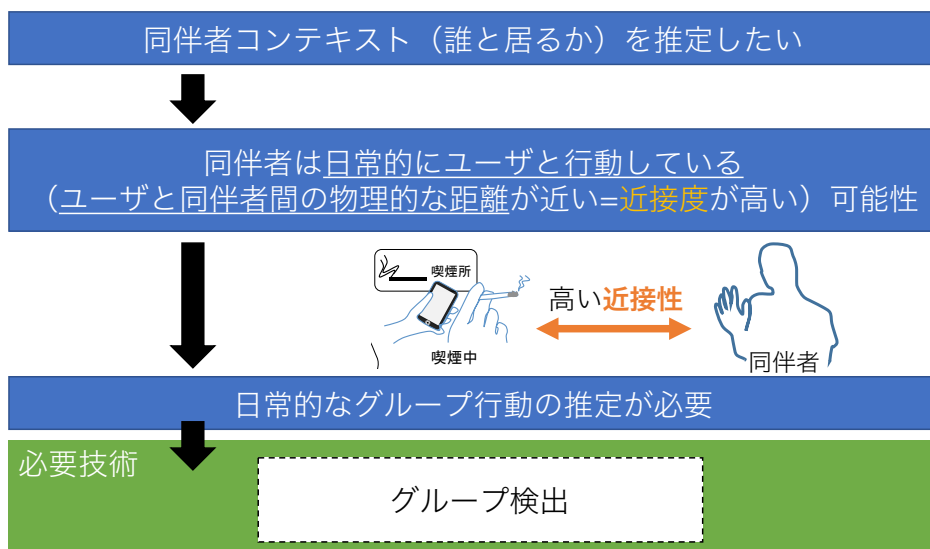


図 3.2: 同伴者コンテキスト推定への必要技術

るには、日常的なグループ行動の推定が必須であると想定される。したがって、日常的なグループ行動を推定するには、室内や建物外において移り変わるグループ(細かい粒度のグループ)を正確に検出する必要がある。

図 3.3 に、本研究における近接度(人々の近さ)とグループの定義について示す。

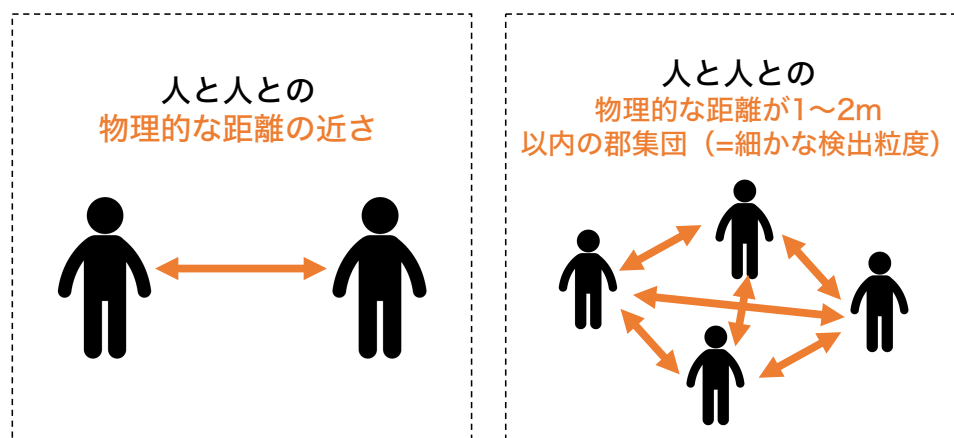


図 3.3: 近接度（人々の近さ）とグループの定義

ここで、同伴者がユーザと日常的に行動している可能性が高い、といった想定から、本研究では近接度（人々の近さ）に着目する．これを人と人との物理的な距離の近さとして定義し、同伴者コンテキスト推定へ利用する．また、本研究で対象とするグループは、人と人との距離が 1 m から 2 m 以内の群集団とする．将来的に本研究で推定したい同伴者コンテキストは、上司なのか、同僚なのか、友人なのか、初対面の人かなどの情報である．このような情報の取得には、長期間に渡るグループ検出によって、どれだけ長い時間、ユーザが誰と一緒に行動したかを、細かい粒度で検出する必要がある．同伴者コンテキストを細かい粒度で取得するには、室内や建物外など、どのような場合においても検出できなくてはならない．

本章では、細かい粒度でのグループ検出を実現するため、スマートフォンに搭載される BLE 信号を計測することで人の近接度を推定し、グループ検出する手法について述べる．

3.2 関連研究

グループ推定に関する研究は、これまで数多く行われている [35, 36, 37, 38, 39, 40, 41, 42, 43]．その中でも特に、代表的なグループ推定の手法として、1) GPS (Global Positioning System) と Wi-Fi を組み合わせて用いた手法、2) BLE (Bluetooth Low Energy) と赤外線センサ (IR) を組み合わせて用いた手法、およ

び3) 環境音や音声を用いた手法, が挙げられる。本節では, 本研究に関連するこれら手法を用いたグループ推定に関する既存研究について詳述する。まず, 3.2.1 項では, グループ推定の既存システムおよび研究について述べる。次に, 3.2.2 項では, 既存のグループ推定手法の課題について述べる。最後に, 3.2.3 項では, 現実に応用され得るグループ推定実現に向けたシステム要件について述べる。

3.2.1 グループ推定の既存システムおよび研究

Jayarajah ら [35] は, 個人の行動が集団のコンテキストによって, どの程度変化するかを確認するため, 個人の移動パターンと集団の移動パターンを統計的に比較している。このシステムのグルーピングの根幹をなすシステムに, GruMon [36] がある。これは Rijurekha らによって開発されたシステムであり, GPS と Wi-Fi のアクセスポイントから取得されるデータから, どのようなグループで建物間を移動しているかを検出するものである。しかし, GPS によるグループ検出は, GPS 衛星を捕捉できる屋外環境においては非常に大きな効果を発揮するが, GPS 衛星の捕捉が困難な屋内環境においては効果的ではない。また, Wi-Fi によるグループ検出は, 同じ建物や同じ部屋に居るということは容易に把握できるが, その中でどのようにグループ化されているかまでは認識できない。従って, 本研究で想定しているオフィスのような区切りのない空間における詳細な(粒度の細かい)グループ検出には適さない。

Trinh-Minh-Tri Do や本研究で着目している BLE は, 周辺機器と端末間などで通信を行う際に用いられる, 近年登場した新たな通信技術であり, BLE を使った研究もなされ始めている [44, 45, 46]。BLE は周囲のデバイスに対して, 直接短いメッセージを送る事ができる上, 端末で利用できるその他の通信技術と比較しても比較的省電力であることが特徴である。そのため, インターネットレス通信の実現手段として大いに期待されている。Trinh-Minh-Tri Do らは, BLE の近接データ (Bluetooth リンクのペア) を基に, 社会的関係におけるグルーピングのための確率モデルを作成している。この研究では, 数ヶ月に渡って集められたオフィスで働く 24 人とそれらの家族 16 人分のデータを利用して性能評価を行い, 約 60% の精度でグループ検出可能なことを示している。

上記研究の拡張方式として、赤外線センサのデータを Trinh-Minh-Tri Do らのモデルに付与することでグルーピングの精度を高め、あらかじめ設定された場所と紐付けることによって、「ミーティング中なのか」、「コーヒープレイク中なのか」、もしくは「チャット中なのか」といった、現在のコンテキスト(状態)を認識するための研究も存在する [37]。しかし、このシステムは、あらかじめ加速度やマイクロフォン、BLE、赤外線センサを搭載した独自開発されたバッジの着用を被験者に義務付けているため、負担がかかる。そのため、実際のサービスとして利用することを考慮した場合、汎用的とはいえない。

岡本ら [38] は、個人が持つノート PC のマイクロフォンを利用して環境音を抽出し、各自の音声をそれぞれのノート PC から取得・認識することで、限られた状況の中で 90% 以上の精度でグルーピングできることを示している。しかし、このシステムは、環境音を利用しているため、雑音等の外乱による影響を極めて受けやすいという欠点がある。また、開発システムをノート PC 上で常に動作させ続ける必要があるため、消費電力を考慮していない。

3.2.2 既存のグループ推定手法の課題

3.2.1 項で述べた既存のグループ推定手法には、解決すべき重大な課題がいくつか存在する。本項では、既存のグループ推定手法の課題点を明らかにし、それらについて以下にまとめる。

- GPS によるグループ推定手法において、人の位置を高精度に補足することは、人と人との物理的な距離(以下、近接度)を正確に把握するのに非常に重要なことである。人の位置を高精度で補足することにより、人の物理的な距離を把握できるため、人の近接度を正確に把握できる。しかしながら、GPS 衛星からの電波を捕捉できない環境(例えば、高層ビル等に囲まれた場所やビル等の屋内環境など)では、位置推定精度が大幅に低下する(位置推定誤差が大きくなる)ため、人の近接度を正確に把握できない。したがって、本研究のような屋内・外問わず比較的高精度かつ細かい粒度でのグループ検出を行う必要がある場合における使用は、非常に難しい。
- Wi-Fi を利用したグループ推定手法では、GPS によるグループ推定手法と

は異なり，比較的高精度で屋内における位置を推定できる．しかしながら，Wi-Fi のアクセスポイントはそれほど密に配置されているわけではないため，人が屋内のどの部屋に居るかなどの大まかな位置情報は推定可能であるが，その部屋のどこに居るかまでの詳細な位置情報は把握できない．そのため，人の物理的な距離が把握できず，正確な近接度を推定できない．また，高精度な位置推定を実現するには，Wi-Fi アクセスポイントを密に配置する必要があり，設置コストが増大する．したがって，オフィスのような区切りのない大きな空間において，粒度の細かいグループ検出には適さない．

- 多種多様なセンサを搭載した特殊デバイスを利用したグループ推定手法では，グループ推定を行うことに特化した独自のデバイスを開発する必要があり，大量生産が難しく，コストもかかる．また，被験者に対して常に特殊デバイスの着用の負担を強いるため，実際のサービスとしての利用を考慮した場合，導入へのハードルが非常に高く，汎用的ではない．さらに，開発された多くの独自デバイスやシステムは，基本的に常時稼働させる必要があるため，消費電力を考慮していない．
- 環境音や音声を利用したグループ推定手法では，雑音の影響を非常に受けやすく，また信号の欠損も非常に多いことから，本研究で想定しているようなグループ検出には適さない．また，音声情報にはプライバシー情報が多く含まれている可能性があるため，プライバシー問題にも適切に対処する必要がある，といった課題がある．

次項において，上記の課題を解決するための要件設定を行う．

3.2.3 要件設定

既存のグループ推定手法の機能特性をまとめた表 3.1 と 3.2.2 項で述べた既存のグループ推定手法の課題点を踏まえ，本項では，現実に応用され得るグループ推定の実現に向けたシステム要件を設定する．本研究において，提案するグループ推定システムは，以下の 4 つの要求を満たす必要がある．

- 検出粒度：屋内・外によらず細かな粒度でのグループ検出が可能である．

表 3.1: 機能比較表

	GPS	Wi-Fi	BLE	IR	音	室内外	独自デバイス	消費電力
Jayarajah, K ら	✓	✓	-	-	-	室外	無	高
Do, T. M. ら	-	-	✓	✓	-	室内	有	低
Okamoto ら	-	-	-	-	✓	室内	無	高
提案手法	-	-	✓	-	-	室外, 室内	無	低

- ユビキタス：商業利用されるスマートフォンのみで実現可能である．それ以外の特殊デバイスは使用しない．
- エネルギー効率：システムは低消費電力である必要がある．
- プライバシー保護：ユーザのプライバシーを損なわない．つまり，個人情報を含むプライバシー情報は取得しない．

本論文では，上記の要件を満たすため，BLE インタフェースを有するデバイスを利用し，人の近接度を計測し，正確なグループ推定を実現する新たなシステムを提案する．

3.3 システム概要

図 3.4 に提案システムの概要を示す．提案システムは，近接度に基づいてグループを検出する．提案システムは，(1) ユーザ間の近接度を計測するためのスマートフォンアプリケーション，および (2) クラウド上でデータを収集するための Firebase，の 2 つの要素から構成される．Firebase は，Google が提供する MBaaS (Mobile Backend as a Service) であり，ユーザ管理や取得データの一元管理をクラウド上で行なっている．また，1 つに集約されたデータを分析し，BLE アドバタイズメントパケット (以下，BLE 信号) を用いてユーザ間の距離を計測する．その後，計測した距離に基づいてグループ検出を行う．グループ検出は，以下の 5 つのステップから構成される．

1. 対象端末同士での BLE 信号の送受信
2. 送受信された各端末データをアップロード

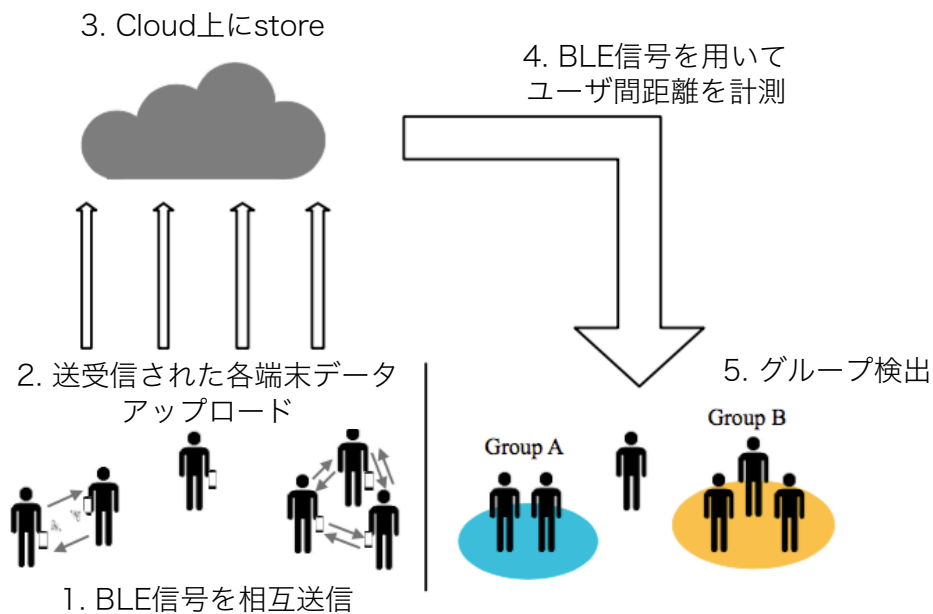


図 3.4: システム概要図

3. Cloud 上に store
4. BLE 信号を用いてユーザ間の近接度を計測
5. グループ検出

本研究では，人の近接度の測定には，BLE を用いる．提案システムでは，ペリフェラルで動作可能な BLE を搭載した Android 端末を対象とし，対象端末で動作するアプリケーションを実現する．BLE は Bluetooth 規格バージョン 4.0 で導入された低消費電力の通信規格であり，端末間で直接的にメッセージの送受信が可能である．そのため，BLE は Wi-Fi とは異なり，細かい粒度で人の近接度を測定できる．また，GPS 衛星からの電波が捕捉できない環境においても，BLE は効果を発揮するため，屋内・外問わず利用可能であり，様々な環境において人の近接度を容易に測定できる．このような理由から，本研究で用いる通信方式として，BLE が適していると考えられる．本節では，まずはじめに，3.3.1 項において，BLE 計測アプリケーションについて述べる．次に，3.3.2 項において，グループ検出手法について述べる．最後に，3.3.3 項において，提案システムの利点について述べる．

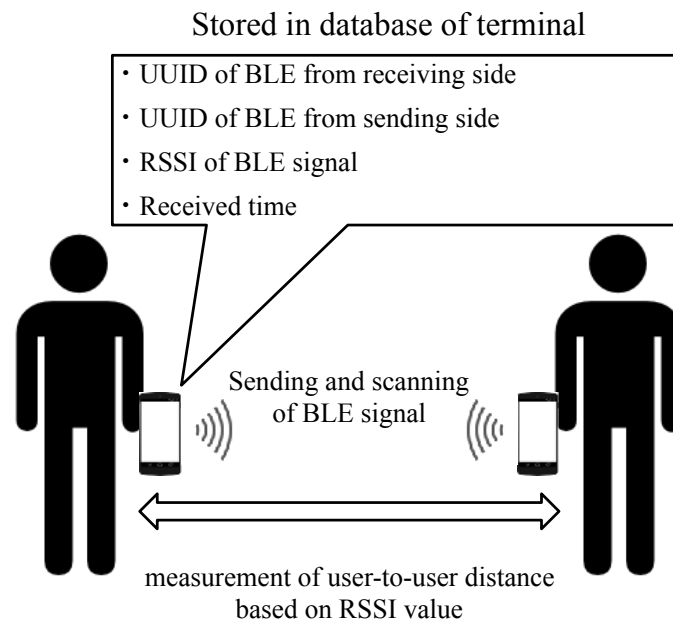


図 3.5: 開発アプリケーション概要図

3.3.1 BLE 計測アプリケーション

本研究で用いるアプリケーションは、ユーザ間の近接度の計測に必要なデータを収集するために作成する。図 3.5 に、開発したアプリケーションの概要を示し、図 3.6 に、開発したアプリケーションの画面構成を示す。

本アプリケーションの機能は、Android 用アプリ開発のため事前に提供されている BLE を対象とした API を用い、BLE 信号を送受信できるように実装する。パケット送信間隔は、API で用意されている 3 段階中で、中間の出力強度を採用している。実際の BLE 信号の送信頻度 [Hz] は公にはされていないため、把握できなかったパケット受信に対して、パケットが端末に到着する度に受信処理（データベースへ保存等）を行うと消費電力への影響が懸念されるため、最小受信間隔を設定する。本研究では、最小受信間隔は 5 秒に設定している。ユーザ間の近接度を測定するため、まずはじめに、端末利用者はアプリケーションを起動させ、開始ボタ

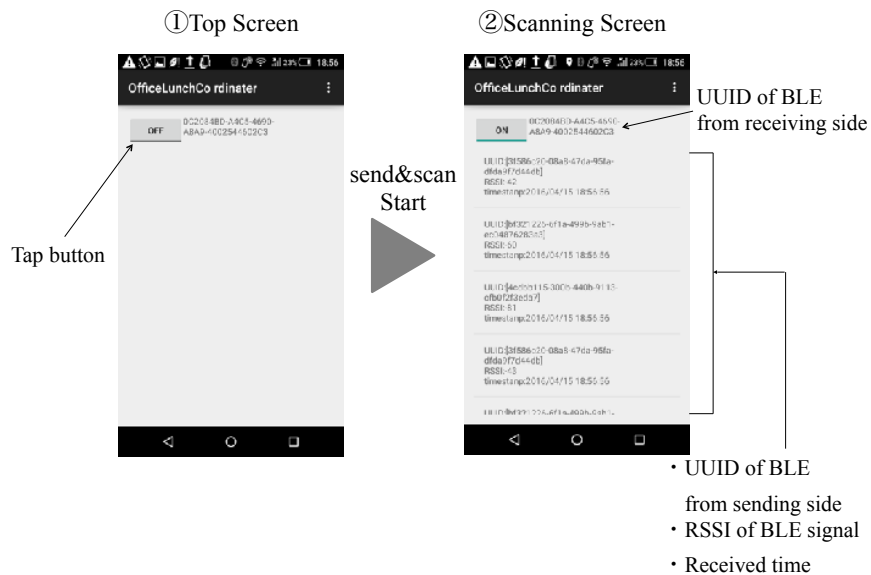


図 3.6: 開発アプリケーションの画面構成

ンを押す．これにより BLE 信号の送信が開始される．次に，周辺端末は送信側が発する BLE 信号を受信すると，以下の 4 つの情報を逐次 Firebase 上のデータベースに格納する．

1. BLE 受信側の UUID
2. BLE 送信側の UUID
3. BLE 信号の RSSI
4. 受信時刻

受信頻度は 5 秒毎であり，常時データ取得するため，バックグラウンドでこれらの各動作を行うように実装している．また，他にも端末利用者がどのようなデータを取得しているか確認するため，送信側の UUID，RSSI，受信日時を画面に表示している．ここで，UUID はユニバーサル固有識別子を意味し，RSSI(Received Signal Strength Indicator) は受信信号強度を意味する．

3.3.2 グループ検出手法

本項では、グループ検出手法について述べる。グループ検出手法では、開発した BLE 計測アプリケーションで取得したデータを使用する。提案するグループ検出手法は、次の 3 つのステップからなる。ステップ 1 では、ユーザが保有する各デバイスから送信された BLE 信号の RSSI 値を収集する。このとき、多くの BLE 信号が同じデバイスから 1 秒毎に受信されるため、雑音が非常に大きくなる可能性がある。そのため、雑音を除去するため、収集された RSSI 値は毎分平均される。ステップ 2 では、ユーザ間の現実的な近接度を測定するため、実験的に設定した RSSI 値の閾値を使用する。本手法では、閾値の決定が極めて重要な要素である。ステップ 3 では、決定された閾値を超える受信データのみを利用してグループ検出を行う。

3.3.3 本システムの利点

本システムは、スマートフォンが唯一必要なデバイスとなる。今までの既存研究で使われる特殊なデバイスと異なり、BLE インタフェースを有する市販スマートフォンがあれば、本システムは実行可能であり、既存システムに比べユビキタスであり、かつ屋内・外によらず適応できる。また、グループ検出を屋内・外で行えるため、本研究が対象とする細かい粒度のグループ（室内や建物外において移り変わるグループ）の検出を可能にする。さらに、コストが非常に低いといった利点も存在する。なぜなら、BLE は低消費電力の通信規格であるため、システムのエネルギー消費が少ない。加えて、本システムはプライバシーを侵害する可能性のある個人情報を取得しないため、プライバシーに考慮している。このように、本システムは既存システムに比べて、多くの利点を有するため、非常に有用である。

3.4 評価実験

提案システムの有効性を示すため、評価実験を行なう。本節では、システム評価のためのデータ収集実験、および収集データを利用したシステム評価結果について述べる。

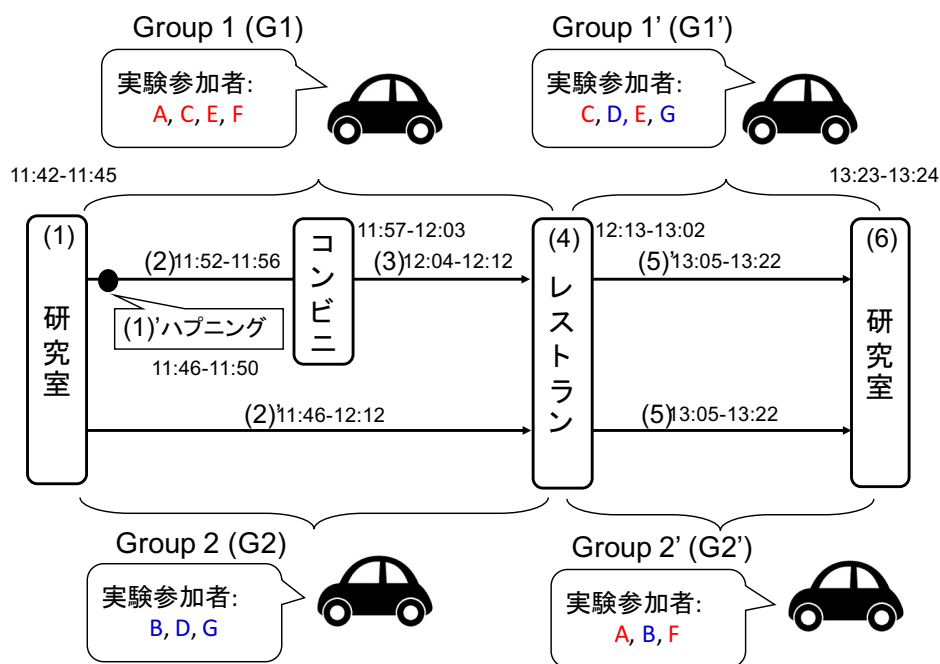


図 3.7: 実験シナリオの全体像

3.4.1 実験の概要

近接度の違いによるグループ検出への影響を調べるため、提案システムを用いて実験を行う。行われた全ての実験において利用されたデバイスは、SHARP SH-03G である。最初に、近接度の推定に利用する RSSI 値の候補閾値を選択するため、BLE 信号の伝搬特性を調べる。調査した BLE 信号の伝播特性にしたがって選択した閾値を用いて、グループ検出への影響を調べる。そして、グループ検出への効果的な閾値を決定する。選択した閾値を用いて、図 3.7 に示すようなシナリオベースの評価を行う。シナリオベースの評価実験では、7 名の被験者（男性：6 名，女子：1 名）に協力を依頼し、各被験者に開発したアプリケーションがインストールされた端末を配布し、実験を行う。本研究では、実験シナリオとして学生のランチ行動を選択し、評価する。学生のランチ行動を選択した理由は、大学生活においてランチ行動は、特に人間関係が如実に現れやすいシチュエーションであると想定し、本システムの検証に適していると考えたためである。また、正解値を収集する

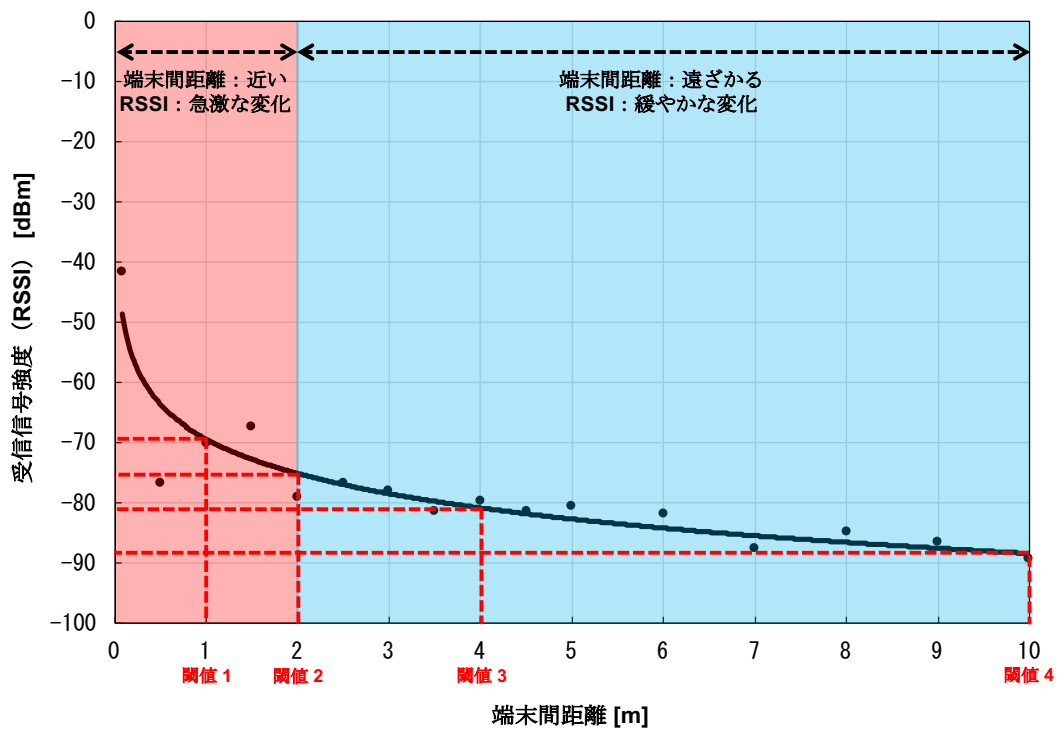


図 3.8: 端末間距離と RSSI 値の関係性

ため、本実験では事前に全被験者に対して“誰と一緒に居たか？”，“何時に何処に行ったのか？”の2つの情報を記録するように要求している．また，記録ミスなどに対応するためのバックアップとして，ライフログアプリである Moves により位置情報等を記録している．次項より，それぞれの実験内容について詳細に述べる．

3.4.2 BLE 信号の伝播特性

被験者間の近接度の測定に用いる RSSI 値の閾値を決定するため，本研究では2つのデバイス間における BLE 信号の電波強度特性（伝播特性）を評価する．本実験では，実験端末に SHARP SH-03G デバイスを2台を使用する．床に実験端末2台を設置し，端末間の距離を変化させながら実験を行う．端末間の距離は，8cm，50cm，100cm，150cm，200cm，250cm，300cm，350cm，400cm，450cm，500cm，

600cm, 700cm, 800cm, 900cm および 1000cm である。上記に設定した距離ごとに 60 秒間、実験端末上でやり取りされるデータを取得する。図 3.8 に実験端末間の距離と RSSI 値の関係を示す。

この図に示される RSSI 値は、60 秒間で受信される BLE 信号の強度の平均値である。図 3.8 を見てわかるように、距離が増加するにつれて RSSI 値が徐々に減少することを示している。特に、端末間の距離が非常に近い 1m~2m までの範囲では、RSSI 値は急激に減少しており、端末間の距離が遠ざかる 2m 以降では、RSSI 値は緩やかに減少していることがわかる。つまり、端末間の距離が近い場合、RSSI 値の変化が急激であることから、近接度の推定が比較的容易であるということを示している一方、端末間の距離が遠くなるにつれて、RSSI の値の変化が緩やかになるため、近接度の推定が難しくなることを示している。この結果から、本研究では、端末間の距離が非常に近い場合の閾値として、-69dBm (距離= 1m) および-75dBm (距離= 2m) を選択し、さらに端末間の距離が遠ざかる場合の閾値として、-82dBm (距離= 4m) および-88dBm (距離= 10m) を選択する。これら 4 つの閾値を用いて、人の物理的な近接度の違いによるグループ検出への影響を調査する。次項では、RSSI 値が決定された閾値を超える受信データのみを利用してグループ検出を行なう。

3.4.3 人の近接度に基づくグループ検出への効果

3.4.2 項の評価で選択した 4 つの閾値を用いて、人の近接度の違いによるグループ検出への影響を調査する。本実験では、図 3.7 の (5), (5)' のデータを調査に用いることとする。ここで、(5) と (5)' はレストランから駐車場への移動を意味する行動である。

図 3.9 の (a) から (d) は人の近接図を表しており、これは Gephi* を用いて描かれている。近接図の閾値は、それぞれ RSSI を -69 dBm, -75 dBm, -82 dBm, -88 dBm として形作られている。図 3.9(a) と図 3.9(b) で示されている RSSI = -69dBm と -75dBm の閾値 (端末間の距離が近い場合) を使用する場合、本手法が正確にグループを検出できることがわかる。また、これらのグループはグラウンド

*Gephi: <https://gephi.org/>

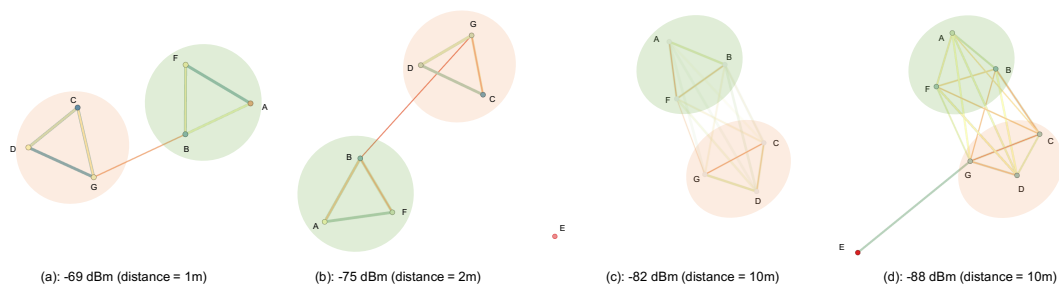


図 3.9: 4 つの RSSI 値での閾値に依存した人の近接図

ツールズと比較し，十分にグループ化されていることが確認できる．図 3.9 はグラントツールズに基づいて各グループを円で囲んだものである．ここで，ピンクの円は $G1'$ を，緑の円は $G2'$ を示している．しかしながら，被験者 E は図 3.9 に示される (a) と (b) の人の近接図に存在しない．その理由は被験者 E は他の実験参加者の近くに居たものの，被験者 E が持つ端末の BLE 信号の伝播出力に問題があったため検出できなかったと考えられる．

一方で，図 3.9(c) と図 3.9(d) に示されているように，RSSI = -82dBm と RSSI = -88dBm の閾値（端末間距離が遠い場合）を用いた場合，2 つのグループが 1 つのグループとしてみなされている可能性があり，本手法はグループ検出を正確に行えないことを示している．その理由は，低い RSSI の値が閾値として使われた場合，遠くにいる被験者も含めた一つの大きなグループが形成されるためである．結果として，遠くの被験者ともコネクションが形成されるためと考えられる．したがって，詳細なグループ検出は，閾値として RSSI = -69 dBm，RSSI = -75 dBm の場合と比較し検出できない．これらの結果から、グループ検出は近接度の違いによって大きく影響されることがわかる。

本実験で対象とするシチュエーションは，学生のランチ行動であり，職場での人間関係と同等に扱うことができると考えられる．人間関係と近接度に関して，Hall[47] は言語や文化によって大きく影響を受けるが，日常での近接度は人間関係によって異なることを示している．Hall の調査では，アメリカ東沿岸における仕事関係での近接度が 1.2m~3.6m と報告されている．これは，本研究で対象とする人の近接度の定義，1m~2m に含まれている．今回は，Hall の調査に基づく仕事関係

の近接度でも中央値付近にあたる 2m を目安として，次項で使用する有効な RSSI 値の閾値として，-75 dBm を選択する．

3.4.4 シナリオに基づく提案手法の評価実験

グループ構成が状況に依存して変化する場合においても，提案システムが正確にグループ検出を行えるかどうかを確認するため，シナリオに基づく提案手法の評価を行う．図 3.7 は本実験でのグループ行動シナリオを示しており，その行動内容は以下のように設定している．

- (1) 研究室から駐車場までの移動
- (1)' 被験者 C が鍵を取りに行くため別行動
- (2) 駐車場からコンビニエンスストアまでの移動
- (2)' 駐車場からレストランまでの移動
- (3) コンビニエンスストアからレストランまでの移動
- (4) レストランでの滞在
- (5) と (5)' レストランから駐車場までの移動
- (6) 駐車場から研究室までの移動

実験の前に，7名の被験者を2つのグループ，被験者 A, C, E, F から構成されるグループ (G1) と，被験者 B, D, G から構成されるグループ (G2) に振り分ける．その時，G1 は図 3.7 に示す，(1)，(1)'，(2)，(3)，(4) の順にシナリオを実行する．一方，G2 は (1)，(2)'，(4) の順でシナリオを実行する．また，レストランから研究室に帰るまでは，2つのグループ，被験者 C, D, E, G から構成されるグループ (G1') と，被験者 A, B, F から構成されるグループ (G2') に振り分け直す．グループの振り分け直しは，グループ構成が変化した場合でも検出可能か確認するためである．本項では，特に (1)，(1)'，(2)，(4) のシナリオに関する結果を示す．

図 3.10 はシナリオ (1) における人の近接図を示している．これは，全実験被験者は研究室から駐車場へ移動している状況である．そのため，被験者らは1つのグループとして検出される．図 3.11 はシナリオ (1)' における人の近接図を表している．この状況では，被験者 C が車の鍵を忘れ研究室へ一時的に取りに帰っており，

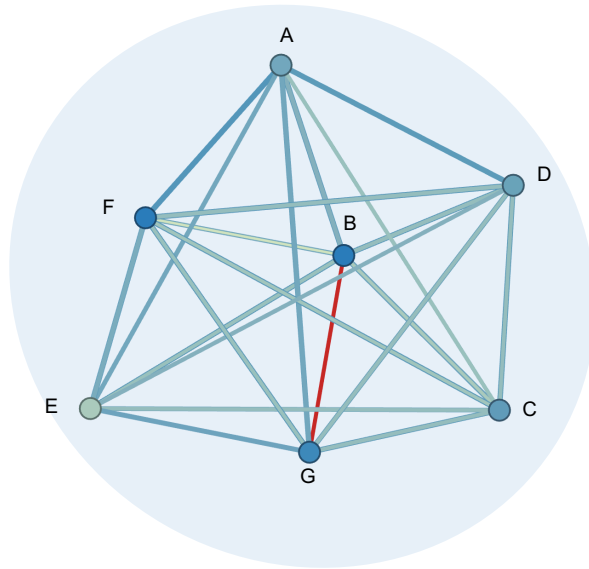


図 3.10: シナリオ (1) における人の近接図

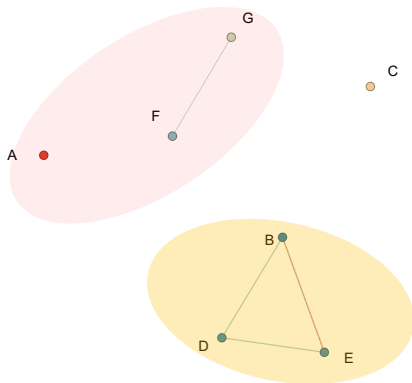


図 3.11: シナリオ (1)' における人の近接図

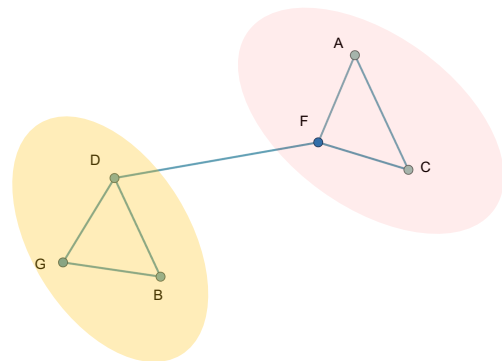


図 3.12: シナリオ (2) における人の近接図

被験者 C が G1 グループの範囲外で検出されていることを示している。この時、被験者 A は G1 グループから少し離れた場所に居たため、被験者 A も G1 グループとして検出されていない。また、G2 グループはこの時、既にレストランへ移動を開始しており、彼らは別グループとして正確に検出されている。図 3.12 はシナリオ (2) における人の近接図を示している。この状況において、各グループは異なる目

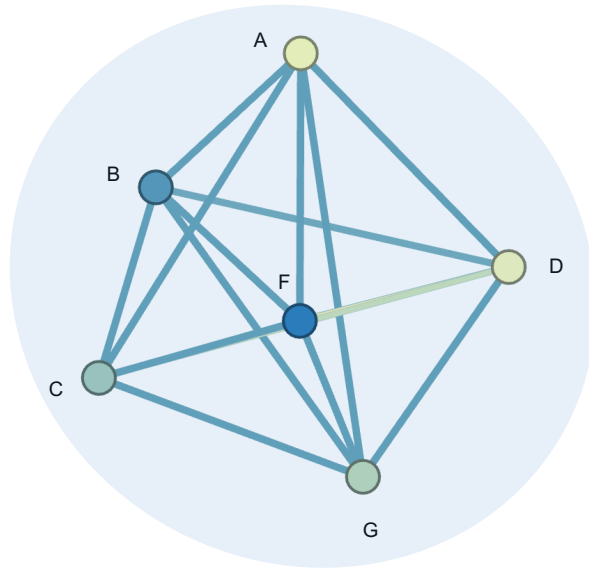


図 3.13: シナリオ (4) における人の近接図

的地へ車で移動している．これより各グループの検出が正確に行われていることがわかる．図 3.13 はシナリオ (4) における人の近接図である．この状況において，全実験被験者がレストランに滞在している．そのため，被験者らは 1 つのグループとして検出されていることを表している．この結果から，提案システムは状況に依ってグループ構成が変化しても正確にグループの検出を行うことができ，またグループ構成が変化する様子も検出できることがわかる．

3.4.5 シナリオに基づくグループ検出率

3.4.4 項に示すシナリオにおいて，提案システムが細かい粒度でグループ検出できるか確認するため，グループ検出率を求める．同伴者コンテキストを正確に推定するには，様々なシナリオ（もしくはシチュエーション）に対応するため，短時間の計測であっても細かい粒度でグループ検出できる必要がある．言い換えれば，細かい粒度でのグループ検出では，時間間隔をなるべく短くしていった場合においても，グループ検出できるかが極めて重要となる．グループ検出を行う際に想定される時間間隔には，1 分，5 分，10 分，30 分，1 時間，1 日など様々考えられる．グ

表 3.2: シナリオごとのグループ検出率

	(1)	(1)'	(2)	(2)'	(3)	(4)	(5)	(5)'	(6)
AG	4	5	6	22	8	50	20	20	2
EG	4	1	6	21	8	50	17	17	2
DG	100.0	20.0	100.0	94.5	100.0	50.0	85.0	85.0	100.0

グループ検出の時間間隔は、シナリオ内容によって変化させる必要があると想定されるが、本実験では細かな粒度でグループ検出できるかを評価するため、最小の時間間隔である 1 分を選択する。ここで、式 (3.41) に評価メトリックとして用いる正解率を定義する。

$$DG = \frac{EG}{AG} \times 100 \quad (3.41)$$

各シナリオで 1 分ごとに分割し、実際に構成されるグループ回数を AG (回)、提案システムでのグループ推定回数を EG (回) とした時、正解率 DG (%) とする。

表 3.2 に、式 (3.41) によって導いた検出率を示す。3.4.4 項に示すシナリオ (1)、(4)、(6) は、被験者全員で構成されるグループなのが特徴で、検出率は 100.0% と高い検出精度を確認できる。3.4.4 項に示すシナリオ (2)、(2)'、(3)、(5) と (5)' は車移動中のシナリオであり、それぞれ、100.0%、94.5%、100.0%、85.0%、85.0% であり、先ほどと同様に高い検出精度とわかる。シナリオ (1)' は、実際に構成されるグループと比較して、正確に検出できた回数は 1 回であり、検出率は 20% にとどまっている。

3.4.6 考察

3.4.5 項で導いた検出率をみると、(2)' を除く全てのシナリオに関して、高い精度でグループ検出できることがわかる。しかし、(2)' のシナリオでは 20% と、他のシナリオよりも低い検出率である。その理由として、今回の定義したグループ構成での近接度が影響していると考えられる。今回の実験では、Hall の調査を参考に、仕事関係の近接度でも中央値付近にあたる $2m$ を目安として、RSSI 値の閾値を -75

dBm とした。そのため、2m よりも遠い距離に居た被験者をグループ構成者として含むことができない。結果として、2m 以上~3m 以内の範囲にいた被験者を除外してしまい、検出率が低下したと考えられる。実際にグループから外された被験者は、検出されたグループから少しだけ離れた場所に位置していた。Hall の調査結果では、仕事関係の人間間の距離間隔は 1.2m~3.2m までであるため、RSSI 値の閾値を変更することで、シナリオ通りに検出できると考えられる。

また、グループ検出する際の時間間隔も、グループ検出精度に影響していると考えられる。今回は細かな粒度でグループ検出できるかを踏まえ、最小の時間間隔である 1 分を選択している。しかし、シナリオ上でのグループを大まかに捉えるには、短い時間間隔でのグループ検出は不向きである可能性がある。実際に (2)' のシナリオでは、1 分毎に検出した結果よりも、3.4.4 項の図 3.11 に示すように、(2)' のシナリオ全体の時間に対してグループ検出を行うほうが、実際のグループ構成に近いグループ検出ができていとわかる。そのため、今回のシナリオでは、1 分間隔でグループ検出を行わずとも、正確にグループ検出ができる可能性がある。

以上より、グループの検出精度は人の近接度の違いや検出する際の時間間隔によって大きく影響を受けると考えられる。したがって、グループ検出において、高い検出精度を達成するには、シナリオに応じた RSSI 値の有効な閾値の選択、および時間間隔の選択が非常に重要となる。有効な閾値と時間間隔を選択することにより、状況の変化に応じてグループの変化を詳細に見ることができる。そのため、長期的なグループ行動に関する情報を蓄積していくことで、より詳細な人間関係を含むグループ推定が実現できる可能性がある。

一方、本研究では、ごく少数の被験者と限られたシナリオ（学生のランチ行動）を用いた評価実験しか行っていないため、多種多様なシナリオを想定した評価実験を今後行う必要がある。例えば、グループ検出に必要な RSSI 値の閾値の決定に関して言えば、グループ人数が 10 人の場合と 100 人の場合では、有効な閾値は異なる可能性が高い。さらに言えば、グループ規模だけでなく、場所の違いなどによってもその閾値は大きく異なる可能性が高い。そのため、効果的な閾値を決定するには、グループの規模やその時のシチュエーションなども考慮して、有効な閾値を動的に決定する必要がある。したがって、それらを決定するための新たなアルゴリズムの考案が必要となると考えられる。

3.5 まとめ

本章では、正確なグループ検出実現のため、BLE 信号によって計測される RSSI 値に依存する人の近接距離に基づくグループ検出システムの提案を行った。本システムは、スマートフォンに搭載された BLE インタフェースから発せられる RSSI 値を計測することで人の近接度を推定し、その近接度に基づいてグループ検出を行うものである。本システムの有効性を示すため、シナリオベースによる性能評価を行なった。その結果、以下のことがわかった。

- 伝搬特性実験において作成した BLE の伝播特性モデルから、グループ検出は人の近接度の違いによって大きく影響されることを確認した。
- 人の近接度に基づくグループ検出への効果を調べた結果、RSSI=-75dBm を人の近接距離を推定するための閾値として選択することにより、正確にグループ検出を行えることを確認した。
- シナリオベース（学生のランチ行動）の実験から、状況に依ってグループ構成が変化してもグループ検出可能であることを確認した。
- シナリオに基づくグループ検出率の評価から、ほとんどのシナリオにおいて高い検出精度を示すことを確認した。

第4章 結論

本研究では、情報通知タイミングに関わるコンテキスト推定を可能にするための基盤システムの実現を目指し、筆者が行った2つの研究について記述した。研究内容は、(1) コンテキスト推定に向けたタッチ操作取得分析システムの実現に関する研究、(2) BLE (Bluetooth Low Energy) を用いたグループ検出に関する手法の研究である。以下、得られた成果と残された課題について述べる。

(1) に関して、タッチ操作の挙動をセンサとして利用できるように、ユーザが行う代表的なタッチ操作の挙動をタッチログから逆解析するアルゴリズムを確立し、このアルゴリズムを実装して、タッチ操作の挙動を認識して取得し、可視化するシステムの開発を行った。開発システムを評価した結果、使用頻度の高い1本指タッチやスワイプ、2本指スワイプは100%、2本指タッチは98%、2本指を用いたピンチイン(アウト)は平均98%、3本~8本までのマルチタッチは平均96.5%の精度で認識できることを確認した。また本システムを用いて、タッチ操作分析によって得られるコンテキストの一つである、スマートフォンの操作形態(持ち方)推定を試みた。Android 端末上で行われる、一般的な8つの操作形態を、機械学習アルゴリズムによって開発された推定モデルで評価した結果、16名の被験者で評価した結果、概ね90%の確率で推定可能な事を確認した。今後は開発した TouchAnalyzer および高精度に推定した操作形態推定手法を応用し、情報通知タイミングに関わるコンテキスト(例えば喫煙中、ゲーム中)を推定する手法の検討、実現が残された課題である。

(2) に関して、BLEによるグループ検出を可能にするため、スマートフォンに搭載されたBLEのRSSI値を取得するシステムを開発し、BLE伝播特性モデルを作成した上で、実験的に最適なしきい値を-75dBmと決定し、グループ検出可能かをシナリオベースの実験により確認した。結果として、提案手法がほぼ正確にグループ検出可能なことを確認した。また、シナリオに基づくグループ検出率を調査した結果、高い検出精度を示すことを確認した。本手法を利用しグループ検出を長期間行なうことで、詳細な人間関係を含むグループ検出が実現できる可能性がある。ま

た人間関係を含むグループ検出が実現できれば、グループの同伴者を考慮して、通知内容を変化させるインテリジェント通知を実現することが可能となる、

残された課題は、実際の長期間でのグループ検出から同伴者コンテキストを推定可能かを検討し、通知内容の判定にまで発展させることである。

謝辞

本研究を進めるにあたり，御多忙の中にも関わらず幾度も議論の時間を設けて頂いた安本慶一教授に，厚く御礼申し上げます．いつも可能性がある限り諦めず，粘り強く，正確に早く何事も処理することの大切さを学ばせていただきました．どんなにご多忙であろうと，研究に対するアドバイスや論文の添削をしてくださる姿は，自身が目指す研究者像の最も根底にあるものと感じております．また学生生活においては，学生である我々が過ごしやすいように，常に我々の事を考えて下さり，その環境をつくろうとしてくださいました．心より感謝申し上げます．

中村哲教授には，御多忙の中にも関わらず論文審査委員を引き受けて頂き，貴重な御意見を賜りました．心より感謝致します．

荒川豊准教授には，本研究テーマを決めるにあたり，自身の意見を鑑みた上で，柔軟な思考で多角的に数多くのアイデアを頂きました．また，本研究を進めるにあたり，的確なご指導および御助言を頂きました．研究に対して真面目である一方，研究旅行の遊びにも全力になる姿に，我々も感化され，全力で遊び，集中して研究をやることの重要性を教えていただきました．また，密に連絡を取れない事があっても，常に学生の事を気にかけて，最後まで研究に対してご助言や他の研究からくる新しいアイデアを頂き，深く御礼申し上げます．

諏訪博彦助教には，本研究を進めるにあたり，的確なご指導および御助言を頂きました．研究の根本，目的は何なのかを常に意識しながら，迷ったときはいつでも暖かく相談に乗っていただき，沢山のアドバイスを頂きました．学生への細かなサポートを熱心にして頂き，心より感謝致します．

藤本まなと助教には，本研究で今までに挑戦したことのなかった，BLE を利用したコンテキスト推定の礎と成る基盤システムの提案，および構築に関する的確なご指導および御助言を頂きました．先生の研究姿勢から，今まで学んで来た知識を別の研究分野へどのように活かすべきかを考える姿勢，および例えうまく行かなくても決してめげることのない心の強さを学びました．学生へ細やかなサポートをして頂き，心より感謝致します．

本学助教を務められておりました，国際電気通信基礎技術研究所 玉井森彦氏には，研究方針や，学会展示会発表，論文の執筆に関して，熱心な御指導や御助言を

頂きました。厚く御礼申し上げます。

金岡恵事務補佐員，山内奈緒事務補佐員，モバイルコンピューティング研究室の尾川恵理事務補佐員には，研究会の手続きや事務処理など等，多くの面で研究生生活を支えて頂きました。また，同性ということも有り，様々なことに対して，気軽に相談させていただきました。謹んで感謝申し上げます。

最後に，実験に御協力頂き，私生活においても大変御世話になった本学ユビキタスコンピューティングシステム研究室の皆様に，御礼申し上げます。

参考文献

- [1] Okoshi, T., Ramos, J., Nozaki, H., Nakazawa, J., Dey, A.K. and Tokuda, H., "*Attelia: Reducing User's Cognitive Load due to Interruptive Notifications on Smart Phones*", Proceedings IEEE International Conference on Pervasive Computing and Communications 2015 (PerCom '15), pp.96–104, ACM (2015).
- [2] Perlow, L.A., "*Administrative science quarterly*", Vol.44, No.1, pp.57–81 (1999).
- [3] Abhinav Mehrotra, Veljko Pejovic, Jo Vermeulen, Robert Hendley, and Mirco Musolesi, "*My Phone and Me: Understanding People's Receptivity to Mobile Notifications*", Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16). ACM, New York, NY, USA, 1021-1032. DOI: <https://doi.org/10.1145/2858036.2858566>
- [4] Abhinav Mehrotra, Robert Hendley, and Mirco Musolesi, "*PrefMiner: mining user's preferences for intelligent mobile notification management*", Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp '16). ACM, New York, NY, USA, 1223-1234. DOI: <https://doi.org/10.1145/2971648.2971747>
- [5] 赤池勇磨, 荒川豊, 諏訪博彦, 安本慶一, "参加型センシングの効率化に向けたコンテキストに基づく応答の推定", 情報処理学会論文誌, Vol.57, No.2, pp.543–552, 2016年2月.
- [6] Juan David Hincapié-Ramos and Pourang Irani. 2013. CrashAlert: enhancing peripheral alertness for eyes-busy mobile interaction while walking. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13). ACM, New York, NY, USA, 3385-3388. DOI: <https://doi.org/10.1145/2470654.2466463>
- [7] Abhinav Mehrotra, Mirco Musolesi, Robert Hendley, and Veljko Pejovic., "*Designing content-driven intelligent notification mechanisms for mobile applications*", Proceedings of the 2015 ACM International Joint Conference

- on Pervasive and Ubiquitous Computing (UbiComp '15). ACM, New York, NY, USA, 813-824. DOI: <https://doi.org/10.1145/2750858.2807544>
- [8] Lau, Sian Lun, and Klaus David. *"Movement recognition using the accelerometer in smartphones."* Future Network and Mobile Summit, 2010. IEEE, 2010.
- [9] Wanmin Wu , Sanjoy Dasgupta , Ernesto E Ramirez , Carlyn Peterson and Gregory J Norman. , *"Classification accuracies of physical activities using smartphone motion sensors."* Journal of medical Internet research 14.5 (2012).
- [10] Ho, J. and Intille, S., *"Using context-aware computing to reduce the perceived burden of interruptions from mobile devices"*, Proceedings of the SIGCHI conference on Human factors in computing systems, ACM, pp. 909– 918 (2005).
- [11] 檜原裕大, 清水裕基, 三好健文, 吉永努, 入江英嗣, "スマートフォンを用いた歩行動作改善ツールの開発", 情報処理学会研究報告, Vol.32, No.7, pp.1–8, 2011年11月.
- [12] Cai, Liang, and Hao Chen. , *"TouchLogger: Inferring Keystrokes on Touch Screen from Smartphone Motion"*, HotSec 11 (2011): 9-9.
- [13] Frank, Mario, et al. , emph"Touchalytics: On the applicability of touch-screen input as a behavioral biometric for continuous authentication.", IEEE transactions on information forensics and security 8.1 (2013): 136-148.
- [14] Hoggan, Eve, et al. , *"Multi-touch rotation gestures: Performance and ergonomics."*, Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. ACM, 2013.
- [15] Harrison, Chris, et al. , *"TouchTools: leveraging familiarity and skill with physical tools to augment touch interaction."*, Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. ACM, 2014.
- [16] Wagner, Julie, Eric Lecolinet, and Ted Selker. , *"Multi-finger chords for hand-held tablets: recognizable and memorable."*, Proceedings of the

- SIGCHI Conference on Human Factors in Computing Systems. ACM, 2014.
- [17] Carl Gutwin, Andy Cockburn, Joey Scarr, Sylvain Malacria, and Scott C. Olson. 2014. Faster command selection on tablets with FastTap. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14). ACM, New York, NY, USA, 2617-2626. DOI=<http://dx.doi.org/10.1145/2556288.2557136>
- [18] Chen, Ke-Yu, et al. , "*uTouch: sensing touch gestures on unmodified LCDs.* ", Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. ACM, 2013.
- [19] Fuccella, Vittorio, Poika Isokoski, and Benoit Martin. , "*Gestures and widgets: performance in text editing on multi-touch capable mobile devices.* ", Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. ACM, 2013.
- [20] Kurosawa, T., Shizuki, B., Tanaka, J. , "*Touch gesture UI focusing on swipe direction on mobile information terminal.* " , IPSJ SIG Technical Report, Vol.2015-HCI-163, No.7, 2015 (in Japanese).
- [21] Goel, M., Wobbrock, J. and Patel, S., "*GripSense: Using Built-in Sensors to Detect Hand Posture and Pressure on Commodity Mobile Phones*", Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology, UIST '12, New York, NY, USA, ACM, pp. 545–554 (online), DOI: 10.1145/2380116.2380184 (2012).
- [22] Markus Löchtefeld, Phillip Schardt, Antonio Krüger, and Sebastian Borning., "*Detecting users handedness for ergonomic adaptation of mobile user interfaces*". Proceedings of the 14th International Conference on Mobile and Ubiquitous Multimedia (MUM '15). ACM, New York, NY, USA, 245-249. DOI=<http://dx.doi.org/10.1145/2836041.2836066>
- [23] Harrison, B. L., Fishkin, K. P., Gujar, A., Mochon, C. and Want, R., "*Squeeze Me, Hold Me, Tilt Me! An Ex-ploration of Manipulative User Interfaces*", Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '98, New York, NY, USA, ACM Press/Addison-

- Wesley Publishing Co., pp. 17–24 (on- line), DOI: 10.1145/274644.274647 (1998).
- [24] Kee-Eung Kim, Wook Chang, Sung-Jung Cho, Junghyun Shim, Hyunjeong Lee, Joonah Park, Youngbeom Lee, and Sangryong Kim., "*Hand grip pattern recognition for mobile user interfaces*", Proceedings of the 18th conference on Innovative applications of artificial intelligence - Volume 2 (IAAI'06), Bruce Porter (Ed.), Vol. 2. AAAI Press 1789-1794.
- [25] 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).
- [26] Taylor, B. T. and Bove, Jr., V. M., "*Graspables: Grasp- recognition As a User Interface, Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*", CHI ' 09, New York, NY, USA, ACM, pp. 917–926 (online), DOI: 10.1145/1518701.1518842 (2009).
- [27] Buschek, D. and Alt, F., "*TouchML: A Machine Learning Toolkit for Modeling Spatial Touch Targeting Behaviour*", Proceedings of the 20th International Conference on Intelligent User Interfaces, IUI ' 15, New York, NY, USA, ACM, pp. 110–114 (online), DOI: 10.1145/2678025.2701381 (2015).
- [28] Buschek, D., Rogers, S. and Murray-Smith, R., "*User-specific Touch Models in a Cross-device Context*", Proceedings of the 15th International Conference on Human-computer Interaction with Mobile Devices and Services, MobileHCI ' 13, New York, NY, USA, ACM, pp. 382–391 (online), DOI: 10.1145/2493190.2493206 (2013).
- [29] 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*", Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '13, New York, NY, USA, ACM, pp. 2795–2798 (online), DOI: 10.1145/2470654.2481386 (2013).
- [30] 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*", Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '13, New York, NY, USA, ACM, pp. 3051–3054 (on-line), DOI: 10.1145/2470654.2481424 (2013).
- [31] Lin, Daw-Tung. , "*Computer-access authentication with neural network based keystroke identity verification.*", Neural Networks, 1997., International Conference on. Vol. 1. IEEE, 1997.
- [32] Monroe, Fabian, Michael K. Reiter, and Susanne Wetzel. , "*Password hardening based on keystroke dynamics.*", International Journal of Information Security 1.2 (2002): 69-83.
- [33] Shanmugapriya, D., and Ganapathi Padmavathi. , "*A survey of biometric keystroke dynamics: Approaches, security and challenges.*" , arXiv preprint arXiv:0910.0817 (2009).
- [34] BERCHTOLD, Martin, et al. Actiserv: Activity recognition service for mobile phones. In: Wearable Computers (ISWC), 2010 International Symposium on. IEEE, 2010. p. 1-8.
- [35] J. Kasthuri, L. Youngki, M. Archan, B. K. Rajesh, "*Need accurate user behaviour?: pay attention to groups!*", Proceedings of UBICOMP '15.
- [36] mei S. Rijirekha, L. Youngki, J. Kasthuri, M. Archan, Krishna and Balan, "*GruMon: Fast and Accurate Group Monitoring for Heterogeneous Urban Spaces*" ,Proceedings of ACM SenSys 2014.
- [37] D. T.Trinh, K. Kyriaki, L. Bruno, P. Fabio, P. G.Daniel, "*Inferring social activities with mobile sensor network*" , Proceedings of ICMI '13.
- [38] M. Okamoto, N. Iketani, K. Nishimura, M. Kikuchi, K. Cho, M. Hattori, and S. Tsuboi, "*Finding Two-level Interpersonal Context: Proximity and Conversation Detection from Personal Audio Feature Data*", Proceedings of INTERSPEECH-2008, pp. 2482-2485, 2008.
- [39] K. Tsubouchi, O. Saisho, J. Sato, S. Araki and M. Shimosaka, "*Fine-grained Social Relationship Extraction from Real Activity Data Under Coarse Supervision*" , Proceedings 2015 ACM International Symposium on Wearable

- Computers, pp. 183-187, 2015.
- [40] Meyer, John P and Herscovitch, Lynne, "*Commitment in the workplace: Toward a general model*", Human resource management review 11.3 (2001): 299-326.
- [41] A. Montanari and S. Nawaz and C. Mascolo and K. Sailer, "*A Study of Bluetooth Low Energy Performance for Human Proximity Detection in the Workplace*", Proceedings of PerCom 2017, pp. 90-99, 2017.
- [42] R. Koshiha and Y. Hirabe and M. Fujimoto and H. Suwa and Y. Arakawa and K. Yasumoto, "*Group Detection Based on User-to-User Distance in Everyday Life for Office Lunch Group Recommendation*", Proceedings of WAINA 2017, pp. 309-314, 2017.
- [43] D. M. T. Trinh, P. G. Daniel, "*GroupUs: Smartphone Proximity Data and Human Interaction Type Mining*", Proceedings of 2011 15th Annual International Symposium on Wearable Computerspp.
- [44] M. Fujimoto, S. Matsumoto, Y. Arakawa, and K. Yasumoto, "*RecurChat: BLE-based Message Forwarding System with On-Site Application Distribution*", Proceedings of ICMU 2016, pp. 122-127, 2016.
- [45] M. Fujimoto, S. Matsumoto, Y. Arakawa, H. Suwa, and K. Yasumoto, "*Development of BLE-based Multi-Hop Communication System for Detecting Slope Failure Using Smartphones*", Proceedings of ICCPPW 2016, pp. 16-21, 2016.
- [46] Partridge, K. and Begole, B., "*Activity-based advertising techniques and challenges*", Proc. Workshop on Pervasive Advertising (2009).
- [47] HALL, E. T. (1963), A System for the Notation of Proxemic Behavior1. American Anthropologist, 65: 1003–1026. doi:10.1525/aa.1963.65.5.02a00020
- [48] 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, 1987, 43.6: 389-395.

業績リスト

主発表

論文誌

1. Yuko Hirabe, Hirohiko Suwa, Yutaka Arakawa, Keiichi Yasumoto, “ TouchAnalyzer: A System for Analyzing User ’ s Touch Behavior on a Smartphone, ” International Journal of Computer Science and Mobile Computing, Volume 7, Issue 1, pp. 25–38, Jan. 2018.

国際会議

1. Yuko Hirabe, Manato Fujimoto, Yutaka Arakawa, Hirohiko Suwa, and Keiichi Yasumoto , “ Effect on Group Detection Based on Human Proximity for Human Relationship Extraction in Daily Life, ” Proceedings of International Workshop on Applications of Wireless Ad-hoc and Sensor Networks (AWASN ’ 17), 6 pages, Aug. 2017.
2. Yuko Hirabe, Yutaka Arakawa and Keiichi Yasumoto , “ Logging All the Touch Operations on Android, ” Proceedings of 7th international conference on mobile computing and ubiquitous networking (ICMU ’ 14), pp. 93-94, Jan. 2014.

国内会議

1. 平部裕子, 諏訪博彦, 荒川豊, 安本慶一, “ タッチ操作ログに基づいた操作形態推定手法, ” 研究報告数理モデル化と問題解決 (MPS) , 2017-MPS-116, No. 9, pp. 1-6, 2017年12月.
2. 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年3月.
3. 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, A0-5, pp. 158-165, 2014年2月.

他発表

論文誌

1. Kazuki Fujisawa, Yuko Hirabe, Hirohiko Suwa, Yutaka Arakawa, Keiichi Yasumoto, “Automatic Live Sport Video Streams Curation System from User Generated Media,” The International Journal of Multimedia Data Engineering and Management (IJMDEM), Vol. 7, Issue. 2, Article. 3, pp. 36-52, April-June, 2016.

国際会議

1. Ryota Koshiba, Yuko Hirabe, Manato Fujimoto, Hirohiko Suwa, Yutaka Arakawa, and Keiichi Yasumoto, “Group Detection Based on User-to-User Distance in Everyday life for Office Lunch Group Recommendation,” The 5th International Workshop on Collaborative Enterprise Systems (COLLABES-2017), pp. 309-314, Mar. 2017.
2. Naoki Maeda, Yuko Hirabe, Yutaka Arakawa, and Keiichi Yasumoto, “COSMS : Unconscious Stress Monitoring System for Office Worker,” The ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp 2016), pp. 329-332, Sept. 2016.
3. Kazuki Fujisawa, Yuko Hirabe, Hirohiko Suwa, Yutaka Arakawa, Keiichi Yasumoto, “Automatic Content Curation System for Multiple Live Sport Video Streams,” The 11th IEEE International Workshop on Multimedia

Information Processing and Retrieval (MIPR 2015), pp. 541-546, Dec. 2014.

国内会議

1. 前田直樹, 平部裕子, 荒川豊, 安本慶一, “ オフィスワークのための自然なストレスモニタリングシステムの提案と設計,” マルチメディア, 分散, 協調とモバイルシンポジウム論文集 (DICOMO2016), 三重県, 2016年7月.
2. 小芝涼太, 平部裕子, 藤本まなと, 長沢忠郎, 岡本昌和, 奥田真也, 瀧塚清孝, 荒川豊, 安本慶一, “ 社内ランチグループ推薦のための日常生活におけるユーザ間距離に基づいたグループ検出,” マルチメディア, 分散, 協調とモバイルシンポジウム論文集 (DICOMO2016), 鳥羽シーサイドホテル, 三重県, 2016年7月.
3. 平部裕子, 荒川周造, 荒川豊, 安本慶一, “ スマートフォンアプリに起因するトラヒックの分析と Zigbee を利用した IoT 機器の通信に対する影響調査,” 情報処理学会 第 79 回モバイルコンピューティングとパーベイシブシステム (MBL) 研究会, 沖縄産業支援センター, 沖縄県, 2016年5月.
4. 宵憲治, 荒川周造, 中村優吾, Pathirathna Shalika, 前田直樹, 松本誠義, 平部裕子, 安本慶一, “ リモコン操作ログと照明演出装置を用いた室内快適化補助システム,” 研究報告コンピュータセキュリティ (CSEC), No.17, pp. 1-8, 2016年2月.