

大学キャンパス内混雑度モニタリングシステムの実装と評価

Implementation and evaluation of a monitoring system for estimating congestion on university campuses

小林 佑太郎[†] 橋本 啓吾[‡] 谷口 義明^{‡,§} 越智 洋司^{‡,§} 井口 信和^{‡,§}
Yutaro Kobayashi Keigo Hashimoto Yoshiaki Taniguchi Youji Ochi Nobukazu Iguchi

1. はじめに

厚生労働省は、新型コロナウイルス感染拡大防止のための取り組みとして、新型コロナウイルス接触確認アプリ COCOA (COVID-19 Contact Confirming Application) [1, 2] を公開している。このアプリは、Apple と Google が提供している Exposure Notification System (ENS) [3] を利用しており、BLE (Bluetooth Low Energy) を利用したソーシャルディスタンス計測により、濃厚接触者の検知や陽性者との接触の通知を可能としている。構成員に COCOA のインストールを義務化している組織もあり、著者らの所属する近畿大学でも教職員や学生に COCOA のインストールを要請している。

大規模大学は、比較的広い地域からさまざまな活動を実施している人が集まること、教室の移動などに伴う人の入れ替わりが激しいことから、感染のリスクが高いと考えられる。近畿大学では感染防止のための様々な取り組みが行われているが、現状、大学内での人の行動や密状態が起きている場所の把握は行われていない。学内の密状態を把握できれば、密環境の改善や学生の行動変容を促す学内インフラ整備、学生登校計画、授業計画改善等に活用できる。

これまでに我々は大学入構者が持つスマートフォンや、大学入構者に配布するビーコン端末を利用して、大学キャンパス内の混雑度をモニタリング、可視化するシステムを提案してきた [4, 5]。提案システムの概要を図 1 に示す。提案システムでは、大学入構者は COCOA が稼働するスマートフォンを保持している、あるいは、COCOA の稼働するスマートフォン

を持っていない場合は入構時に貸し出された BLE ビーコン端末を保持していることを想定する。そのため、大学入構者は新たにスマートフォンアプリを導入する必要がない。また、提案システムでは、学内の複数個所にモニタリング端末が設置されることを想定する。モニタリング端末では、周辺に存在する COCOA の稼働するスマートフォンや BLE ビーコン端末からブロードキャスト送信される BLE アドバタイジングパケットをモニタリングする。モニタリング端末で観測された情報は学内無線 LAN を介してサーバに集約され、サーバはエリアごとの混雑度を管理者や大学入構者にフィードバックする。しかし、文献 [4, 5] では、提案システムの機能うち一部のみの実装、評価にとどまっていた。

本稿では、我々の提案するシステム全体の実装と評価を行う。実装システムでは、モニタリング端末として Raspberry Pi 4 を、サーバとして一般的な構成の Linux PC を用いる。実装システムでは、ネットワーク接続に近畿大学全学無線 LAN を利用していることと、Raspberry Pi 4 のみでモニタリング端末を構成できることから、Raspberry Pi 4 を購入、設定、設置するだけで容易に学内のモニタリング箇所を増やすことができる。本稿では、研究室内の小規模実験により実装システムの基本評価を行う。

以降の本稿の構成は以下のとおりである。2 章で関連研究について述べ、3 章で提案システムについて述べる。4 章で提案システムの実装について述べ、5 章でシステムの評価を行う。最後に 6 章でまとめと今後の課題を述べる。

2. 関連研究

新型コロナウイルスへの対策としてカメラ [6, 7]、スマートフォン [1, 2, 8]、独自デバイス [9-12] 等を用いてソーシャルディスタンス計測を行うシステムがある。我々のグループでも、文献 [11, 12] において、組み込み機器を用いることによりソーシャルディスタンスを計測するシステムを提案している。一方、本研究では、ソーシャルディスタンスではなく特定エリアの混雑度を推定することを目的としている。

特定エリアの人数や混雑度を推定する研究は古くから行われている。混雑度推定のためには、カメラ [13, 14]、センサ [15-17]、WiFi プローブ要求信号 [18, 19]、WiFi の受信電波強度やチャンネル状態情報 [20-22] などが使われる。我々の提案する混雑度モニタリングシステムでは、COCOA からブロードキャスト送信される BLE アドバタイジングパケットを利用するため、モニタリングのために新たにスマートフォンアプリケーションを導入する必要がない。

大学内を対象とした混雑度モニタリングに関する研究はいくつかある。大阪市立大学、新潟大学、横浜国立大学などの大学では各座席に貼りつけた QR コードを用いて着席状況をモニ

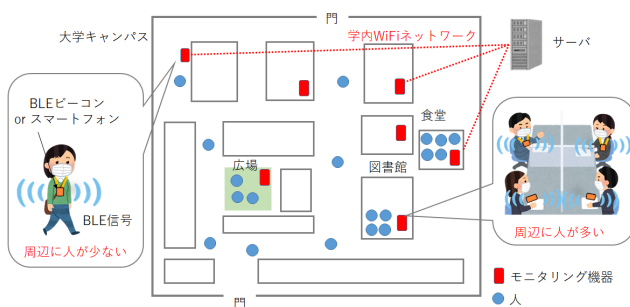


図 1 提案モニタリングシステム [4, 5]

[†] 近畿大学大学院総合理工学研究科, Graduate School of Science and Engineering, Kindai University

[‡] 近畿大学理工学部, Faculty of Science and Engineering, Kindai University

[§] 近畿大学情報学研究所, Cyber Informatics Research Institute, Kindai University

タリングするシステムを開発、運用している。学生が自身の持つスマートフォン等により QR コードを読み込むことにより、着席状況をモニタリングする。大きなコストをかけることなく混雑度に関する情報をモニタリング可能であるが、QR コードの貼付けや読み取りが難しような時間帯や場所での利用は困難と考えられる。

静岡理科大学では、Wi-Fi アクアスポイントで収集される情報から自動的に教室毎の混雑度把握を行うシステムを開発、運用している [23]。このシステムでは、追加の機材を導入することなく混雑度を推定できるが、Wi-Fi に接続されている端末の情報しか取得できない。また、人が Wi-Fi 端末を複数台持っている場合や持っていない場合も考えられる。

東京大学では、新型コロナウイルス対策のためのスマートフォンアプリ MOCHA (Mobile CHeck-in Application) [24] を開発している。このアプリでは COCOA を補完する機能が導入されており、学内混雑状況の可視化や学内公共スペースの滞在予約、事前許可したユーザー間での位置情報の共有などが可能である。これに対して、本稿で提案するシステムでは、学内の密状態の把握および可視化を主な目的としており、また、新たなスマートフォンアプリケーションを導入する必要がない。

九州大学では、学生が持つモバイル端末から送信される WiFi の Probe Request 信号と COCOA の送信する BLE パケットを利用して混雑度を推定する手法を検討している [25]。ただし、人が Wi-Fi 端末を複数台持っている場合や持っていない場合も考えられる。また、WiFi Probe Request 信号の情報と COCOA の送信する BLE パケットの情報をどのように組み合わせるかが課題となると考えられる。本研究では、COCOA の送信する BLE パケットのみを用いるため、より単純な仕組みで混雑度推定が可能となると考えられる。

3. 大学キャンパス内混雑度モニタリングシステム

本章では、我々の提案する大学キャンパス内混雑度モニタリングシステム [4,5] について述べる。本研究では、大学入構者は COCOA が稼働するスマートフォンを保持していることを想定する。COCOA は定期的に BLE アドバタイジングパケットをブロードキャスト送信しており、提案システムではこれを混雑度推定に用いる。なお、保有するスマートフォンが COCOA の要求スペックを満たさないなど、COCOA の稼働するスマートフォンを持っていない場合、入構時に入構許可証として BLE ビーコン端末を貸与するものとする。BLE ビーコンも COCOA と同様に定期的に BLE アドバタイジングパケットをブロードキャスト送信する。BLE ビーコン端末は首からかけることを想定する。

提案システムの概要を図 1 に示す。提案システムでは、学内の複数個所にモニタリング端末を設置する。モニタリング端末の設置場所としては、食堂やオープンスペース、図書館など密状態が起きやすいと考えられる場所を想定する。モニタリング端末は、周辺に存在する COCOA の稼働するスマートフォンや BLE ビーコン端末からブロードキャスト送信される BLE アドバタイジングパケットを受信し、その受信時刻と受信電波強度、MAC アドレスなどのパケット内に含まれる情報を記録する。これらのモニタリング端末で取得された情報は、一定間隔ごとに学内無線 LAN を介してサーバに集約される。サーバ

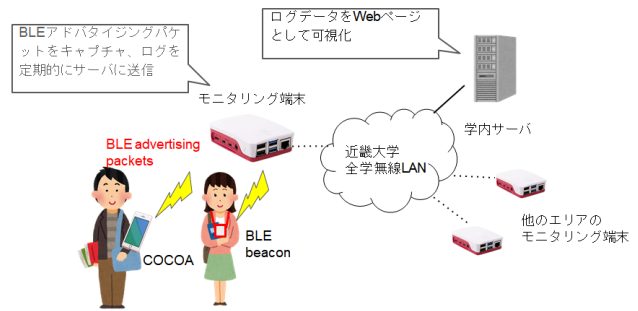


図2 実装システムの構成

では取得したデータからエリアごとの混雑度を推定し、管理者や大学入構者にフィードバックする。

4. システムの実装

本章ではシステムの実装について述べる。

4.1 システム構成

実装システムの構成を図2に示す。本研究では、学内ネットワークに接続した標準的なLinux PC (OS: CentOS 8) をサーバとして用いた。サーバの処理のほとんどはLinuxの標準的なコマンドのみで実現しており、様々な環境で動作すると考えられる。また、サーバにはApache 2.4.41をインストールし、混雑度情報をWebページとして閲覧できるようにした。

モニタリング端末としてはRaspberry Pi 4 (OS: Raspbian GNU/Linux 10 (buster)) を用いた。実装システムでは、モニタリング端末は、BLE アドバタイジングパケットを受信しやすいよう、なるべく上部に設置することを想定する [4,5]。BLE アドバタイジングパケットのキャプチャにはRaspberry Pi 4内蔵のBluetoothモジュールを、データのサーバへの転送にも内蔵のWiFiモジュールを用いており、Raspberry Pi 4に追加のモジュールを追加することなくモニタリング端末として稼働可能である。モニタリング端末ではBLE アドバタイジングパケットをキャプチャするためにPythonおよびbluepyライブラリをインストール、設定した。また、Raspberry Pi標準ケースにマグネットを取り付け、壁に取り付けられるようにした。2021年7月時点で、近畿大学東大阪キャンパス38号館、31号館、アカデミックシアター、情報処理教育棟(KUDOS棟)の一部にモニタリング端末を設置している。モニタリング端末の外観と実際の取付例を図3に示す。

BLE ビーコンとしては、iBeaconに対応した安価なBLE ビーコンである丸紅情報システム株式会社製のBlueBeacon Tag SE [26] (サイズは31×30×9.3 mm)を用いた。本BLE ビーコンの外観を図4に示す。BLE ビーコンはストラップをつけて首からかける、入構許可証、教職員証等のストラップと一緒に装着することを想定する。本BLE ビーコンは一定間隔でBLE アドバタイジングパケットをブロードキャスト送信する。実装システムでは本ビーコンを初期設定のまま用いた。BLE アドバタイジングパケットに含まれるUUID (Universally Unique Identifier) は全てのBLE ビーコンで同じであり、iBeaconパケットのMinor値にはBLE ビーコンごとに異なる識別番号が含まれる。また、パケットの送信強度は-4 dBm、BLE アドバタイジングパケットの送信間隔は1秒



(a) モニタリング端末外観（裏と表）



(b) 設置の様子（KUDOS 棟）



(c) 設置の様子（アカデミックシアター）

図 3 モニタリング端末

である。本設定での本 BLE ビーコンの想定電池寿命は 1 年である。

4.2 モニタリング端末からの情報収集処理

モニタリング端末の情報収集にかかわるプログラムの構成を図 5 に示す。本システムはサーバ、モニタリング端末共に Linux ベースの OS が動作している。したがって、Linux で利用されているソフトウェアやライブラリを利用してシステムを実装した。



図 4 BLE ビーコン

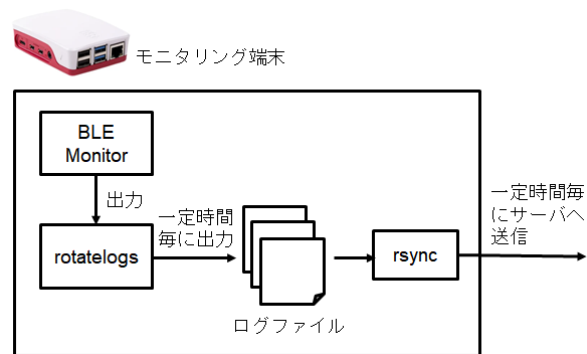


図 5 モニタリング端末の情報収集に関するプログラムの構成

モニタリング端末において BLE アドバイジングパケットをキャプチャするために Python および bluepy ライブラリ [27] を用いてモニタリング用プログラム（図中、BLE Monitor）を作成した。モニタリング用プログラムは、モニタリング端末起動時に自動実行され、内蔵 BLE モジュールで受信される BLE アドバイジングパケットを常にチェックする。受信したパケット中に含まれる UUID が COCOA で使用される UUID あるいは BLE ビーコンで使用される UUID と一致する場合、そのパケットの受信時刻、受信電波強度および送信元 MAC アドレスを出力する。

モニタリング用プログラムの出力は rotatelog [28] に渡され、一定時間ごとにログファイルとして保存される。rotatelog は Apache のログを保存するために開発されたプログラムであり、Apache のユーティリティパッケージに含まれる。rotatelog を使うと一定時間ごとにファイルを分けてログを保存することができる。実装システムでは、ファイルの保存間隔を 10 分としている。

モニタリング端末は近畿大学全学無線 LAN に接続されており、cron により、一定時間間隔で rsync を用いて保存したログファイルをサーバに送信するよう設定されている。実装システムでは、この間隔を 10 分としている。また、モニタリング端末は NTP (Network Time Protocol) により学内 NTP サーバと時刻同期を行うよう設定している。なお、近畿大学東大阪キャンパスでは、全学無線 LAN としてキャンパス全域をカバーするように約 1000 台のアクセスポイントが設置されてい

る。そのため、今後、学内のさまざまな場所にモニタリング端末を設置することが可能である。

4.3 サーバにおける混雑度推定に関する処理

サーバでは一定時間間隔でモニタリング端末から受信したログファイルに基づきモニタリング端末付近の混雑状況を推定する。実装システムではこの間隔を 10 分にしている。本研究では混雑度を推定するためのプログラム（混雑度推定プログラム）をシェルスクリプトを使用して作成し、cron により定期的に実行されるように設定した。

混雑度推定プログラムは、モニタリング端末から送信されたログファイルに基づき、各部屋の混雑度を推定する。なお、モニタリング端末の集合を $\mathcal{N} = \{N_1, N_2, \dots\}$ と表記する。また、モニタリング端末の設置場所は事前に与えられるものとする。プログラムはまず、COCOA、BLE ビーコンの受信ログから、BLE アドバイジングパケットの送信元 MAC アドレスを取得する。ここで、モニタリング端末 N_i において受信した COCOA の送信元 MAC アドレスの集合を $\mathcal{M}_{C,i}$ 、BLE ビーコンの送信元 MAC アドレスの集合を $\mathcal{M}_{B,i}$ 、全 MAC アドレスの集合を $\mathcal{M}_i = \mathcal{M}_{C,i} \cup \mathcal{M}_{B,i} = \{M_1, M_2, \dots\}$ と表記する。また、モニタリング端末 N_i において MAC アドレス M_j を持つ端末から送信された BLE アドバイジングパケットの平均受信電波強度を $\bar{r}_{i,j}$ 、パケット受信回数を $p_{i,j}$ と表記する。我々の過去の予備実験によると、部屋の外など遠くに位置する端末からの BLE アドバイジングパケットが届くことがあった [4, 5]。そこで、実装システムでは、下記のように、COCOA、BLE ビーコンそれぞれについて平均受信電波強度およびパケット受信回数が一定以上の MAC アドレスの集合を取得し、混雑度推定に用いる。

$$\begin{aligned} \mathcal{M}'_{C,i} &= \{M_j \in \mathcal{M}_{C,i} \mid \bar{r}_{i,j} \geq R_C, p_{i,j} \geq P_C\} \\ \mathcal{M}'_{B,i} &= \{M_j \in \mathcal{M}_{B,i} \mid \bar{r}_{i,j} \geq R_B, p_{i,j} \geq P_B\} \end{aligned}$$

ここで、 R_C 、 P_C 、 R_B 、 P_B は閾値であり、事前実験により求めるものとする。

なお、モニタリング端末が密に設置されている場合は、同じ MAC アドレスの BLE アドバイジングパケットを複数のモニタリング端末が受信する可能性がある。そのような場合には、最も高い受信電波強度でパケットを受信したモニタリング端末でのみパケットを受信したものとして扱う。例えば、MAC アドレス M_k を持つ端末のパケットが複数のモニタリング端末で受信され、モニタリング端末 N_m において最も高い受信電波強度でそのパケットを受信したとする。この時、モニタリング端末 N_m の MAC アドレスの集合 \mathcal{M}'_m にはそのアドレスを残し、他のモニタリング端末の MAC アドレスの集合からは M_k を除去する。

上記の処理の後、モニタリング端末 N_i の周辺の COCOA が稼働する端末数を $n_{C,i} = |\mathcal{M}'_{C,i}|$ 、BLE ビーコン数を $n_{B,i} = |\mathcal{M}'_{B,i}|$ として算出する。本研究では簡単のため、これらの和によって、モニタリング端末 N_i の周辺の混雑度を下記の 3 種類で表す。

$$\text{混雑度} = \begin{cases} \text{小} & (n_{C,i} + n_{B,i} < N_1) \\ \text{中} & (N_1 \leq n_{C,i} + n_{B,i} < N_2) \\ \text{大} & (N_2 \leq n_{C,i} + n_{B,i}) \end{cases} \quad (1)$$

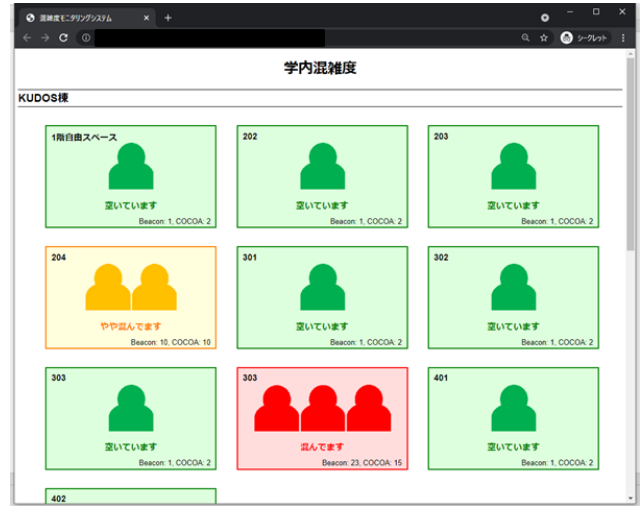


図6 Web ページ例

ここで、 N_1 、 N_2 は混雑度を決める閾値である。

4.4 サーバにおける可視化処理

サーバでは前節で説明した混雑度推定処理を実施した後、処理結果に基づいて Web ページを生成する。本研究では、Web ページを生成するためのプログラムをシェルスクリプトを使用して作成した。

サーバが生成する Web ページの例を図 6 に示す。混雑度に応じて、緑、黄色、赤と色をわけて表示している。学内から閲覧する場合は直接サーバに接続することにより本 Web ページを閲覧できる。また、近畿大学理工学部情報学科が管理する公開用 Web サーバ上でリバースプロキシの設定をしており、学外から情報学科の Web サーバ経由で本 Web ページを閲覧することも可能である。

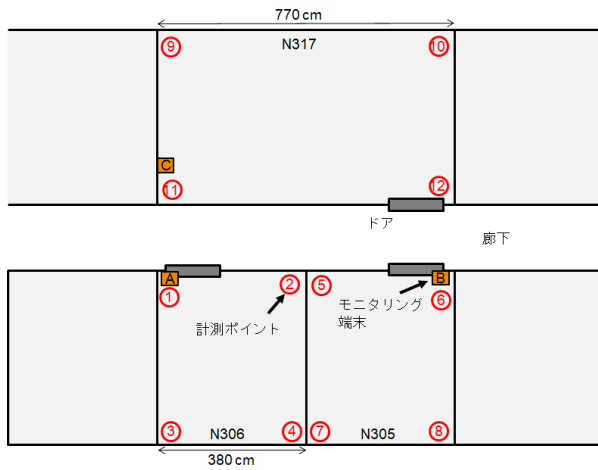
5. システムの基礎評価

5.1 実験環境

実験は近畿大学東大阪キャンパス 38 号館 3 階知的通信網研究室で行った。実験では、図 7 のように、N317、N305、N306 の 3 部屋の各部屋にモニタリング端末を設置し、パケット計測を行った。モニタリング端末の設置場所は、近くに電源があること、什器等の影響なく設置が容易なこと、他の部屋に展開するときと同じ条件で設置できることなどを考慮し、N305、N306 は入口のドア上に設置した。N317 は什器等の配置の関係上、壁際に設置したポール上にモニタリング端末を設置した。また、各部屋の 4 隅、合計 12 か所（図中、1～12 で示される設置箇所）に BLE ビーコンおよび COCOA の稼働するスマートフォン（Sharp AQUOS sense 3 H-M12X5-B、OS: Android 9）を設置し、それぞれの設置箇所ですら 3 分間パケットを観測した。

5.2 実験結果

表 1、表 2 は、BLE ビーコンおよび COCOA の稼働するスマートフォンからの BLE アドバイジングパケットの計測結果である。表中、最も受信電波強度が高く検出される箇所を太字で示している。ND は検出されなかったことを表す。表より、端末からの BLE アドバイジングパケットを送信する場



(a) 実験環境



(b) N305



(c) N306



(d) N317

図7 実験の様子

合、同じ部屋にあるモニタリング端末で計測された場合に最も受信電波強度が高くなることがわかる。N305 と N306 は隣接する部屋であり、また廊下側にモニタリング端末を設置しているため、部屋内の廊下に近い位置に端末を置いた場合には、他の部屋でもやや高い受信電波強度で BLE アドバタイジングパケットを受信する可能性があるが、同じ部屋のモニタリング端末の方が高い受信電波強度でパケットを受信する。

今回の実験では端末が静止した状態での評価を行った。端末が動く場合には、モニタリング端末に近づいた時だけ BLE アドバタイジングパケットが計測されると考えられる。そのような場合に、パケット受信回数を使うことにより、その場にずっと滞在しているのか、通りかかっただけなのかを判定することができると考えられる。そのような端末が動く場合の評価については今後の課題とする。

6. まとめと今後の課題

本稿では、我々がこれまでに提案してきた大学キャンパス内混雑度モニタリングシステムの実装と評価を行った。本システムでは大学内に設置したモニタリング端末が、周辺のスマートフォン上の COCOA から送信される BLE アドバタイジングパケットや BLE ビーコンから送信される BLE アドバタイジングパケットを観測することにより、周辺の混雑度を推定す

る。また、モニタリング端末で観測した情報をサーバに収集、解析し Web ページとして出力することにより、学内の混雑度を可視化する。実装システムでは、モニタリング端末として Raspberry Pi 4 を、サーバとして一般的な構成の Linux PC を用いた。実装システムでは、Raspberry Pi 4 を購入、設定、設置するだけで容易に学内のモニタリング箇所を増やすことができる。本稿ではシステムの基本評価を行った。

今後の課題として、現在では各部屋に 1 台ずつモニタリング端末を設置する場合の実装、評価しか行っていない。今後、各部屋や空間に複数台のモニタリング端末を設置してより精度の高いモニタリングを実施することを考えている。また、得られたデータを地図上に表示するなど視覚的にわかりやすく可視化することも考えている。さらに、これらのデータを使って混雑度予測や適切なフィードバックを行い学生の行動変容を促すための仕組みも検討する必要がある。

謝辞

本研究の一部は“オール近大”新型コロナウイルス感染症対策支援プロジェクト、令和 2 年度近畿大学学内研究助成金 (SR08) および科学研究費 (課題番号 19K11934) の補助を受けている。また、機材設置においてご協力いただいた近畿大学石井雅基氏、辻諭氏に感謝する。

表1 BLE ビーコンからの BLE アドバタイジングパケットの計測結果

設置箇所	計測箇所 A (N306)		計測箇所 B (N305)		計測箇所 C (N317)	
	平均 RSSI [dBm]	受信回数	平均 RSSI [dBm]	受信回数	平均 RSSI [dBm]	受信回数
N306 1	-71.2	32	-74.4	71	-86.7	30
N306 2	-68.2	48	-84.9	74	-83.6	52
N306 3	-80.4	33	-88.9	57	ND	ND
N306 4	-80.8	49	-86.0	55	-90.0	6
N305 5	-77.6	43	-69.8	47	-87.0	1
N305 6	-78.9	31	-65.7	57	-91.3	10
N305 7	-82.3	19	-69.2	55	ND	ND
N305 8	-86.6	20	-79.4	32	ND	ND
N317 9	ND	ND	ND	ND	-74.0	64
N317 10	ND	ND	-90.6	3	-80.0	42
N317 11	-93.9	11	-86.7	37	-57.5	36
N317 12	-89.0	2	-81.4	33	-74.4	44

表2 COCOA の稼働するスマートフォンからの BLE アドバタイジングパケット計測結果

設置箇所	計測箇所 A (N306)		計測箇所 B (N305)		計測箇所 C (N317)	
	平均 RSSI [dBm]	受信回数	平均 RSSI [dBm]	受信回数	平均 RSSI [dBm]	受信回数
N306 1	-76.6	42	-89.6	120	-97.6	5
N306 2	-86.3	81	-89.1	138	ND	ND
N306 3	-85.1	45	-85.1	98	-98.0	3
N306 4	-82.6	103	-92.0	112	ND	ND
N305 5	-79.3	89	-69.1	205	-80.4	116
N305 6	-80.7	92	-69.4	220	-82.3	93
N305 7	-86.4	103	-80.6	188	ND	ND
N305 8	-89.8	41	-82.0	189	ND	ND
N317 9	ND	ND	ND	ND	-71.6	221
N317 10	ND	ND	ND	ND	-85.2	78
N317 11	ND	ND	ND	ND	-68.1	222
N317 12	ND	ND	-94.6	ND	-79.0	86

参考文献

- [1] 厚生労働省, “新型コロナウイルス接触確認アプリ (COCOA) COVID-19 Contact-Confirming Application,” Jun. 2020, https://www.mhlw.go.jp/stf/seisakunitsuite/bunya/cocoa_00138.html.
- [2] 新型コロナウイルス感染症対策テックチーム, “接触確認アプリ及び関連システム仕様書,” May 2020, <https://cio.go.jp/node/2613>.
- [3] Apple and Google, “Exposure Notification Bluetooth Specification,” Apr. 2020, <https://www.blog.google/documents/62/Exposure-Notification-Bluetooth-Specification.v1.1.pdf>.
- [4] Y. Taniguchi, T. Mukaida, Y. Ochi, and N. Iguchi, “A BLE-based monitoring system for estimating congestion on university campuses,” in *Proceedings of IEEE LifeTech 2021*, Mar. 2021, pp. 418–419.
- [5] 向田朋樹, 谷口義明, 越智洋司, 井口信和, “BLE アドバタイジングパケットを利用した大学キャンパス内の混雑度モニタリングシステム,” *電子情報通信学会技術研究報告*, Jan. 2021.
- [6] アースアイズ, “ソーシャルディスタンスカメラ,” 2020, <https://earth-eyes.co.jp/sdc/>.
- [7] N. S. Punn, S. K. Sonbhadra, and S. Agarwal, “Monitoring COVID-19 social distancing with person detection and tracking via fine-tuned YOLO v3 and Deepsort techniques,” *arXiv:2005.01385*, May 2020.
- [8] MAMORIO, “Mamorio,” 2020, <https://mamorio.jp/>.
- [9] Singapore Government, “TraceTogether Token: Media Statement,” 2020, <https://www.tech.gov.sg/media/media-releases/2020-06-16-tracetgether-token-media-statement>.
- [10] IK Multimedia, “Safe spacer,” 2020, <https://www.safespacer.net/>.
- [11] 小林佑太郎, 谷口義明, 多田昌裕, 波部斉, 越智洋司, 溝渕昭二, 半田久志, 井口信和, “小型無線端末を用いた大学内ソーシャルディスタンスモニタリングシステムの検討,” *2020年度情報処理学会関西支部支部大会講演集*, pp. 1–7, Sep. 2020.

- [12] Y. Kobayashi, Y. Taniguchi, Y. Ochi, and N. Iguchi, “A system for monitoring social distancing using microcomputer modules on university campuses,” in *Proceedings of IEEE ICCE-ASIA 2020*, Nov. 2020, pp. 1–4.
- [13] A. B. Chan and N. Vasconcelos, “Counting people with low-level features and Bayesian regression,” *IEEE Transactions on Image Processing*, vol. 21, no. 4, pp. 2160–2177, Apr. 2012.
- [14] Z. Ma and A. B. Chan, “Crossing the line: Crowd counting by integer programming with local features,” in *IEEE CVPR 2013*, Jun. 2013, pp. 2539–2546.
- [15] F. Bu, R. Greene-Roesel, M. C. Diogenes, and D. R. Ragland, “Estimating pedestrian accident exposure: Automated pedestrian counting devices report,” *UC Berkeley Traffic Safety Center*, pp. 1–28, Mar. 2007.
- [16] R. Greene-Roesel, M. C. Diogenes, D. R. Ragland, and L. A. Lindau, “Effectiveness of a commercially available automated pedestrian counting device in urban environments: comparison with manual counts,” *TRB 2008 Annual Meeting*, pp. 1–16, 2008.
- [17] S. Fujii, Y. Taniguchi, G. Hasegawa, and M. Matsuoka, “Pedestrian counting with grid-based binary sensors based on Monte Carlo method,” *SpringerPlus*, vol. 3, pp. 1–10, Jun. 2014.
- [18] Y. Wang, J. Yang, H. Liu, Y. Chen, M. Gruteser, and R. P. Martin, “Measuring human queues using WiFi signals,” in *Proceedings of ACM MobiCom 2013*, Sep. 2013, pp. 235–237.
- [19] L. Schauer, M. Werner, and P. Marcus, “Estimating crowd densities and pedestrian flows using Wi-Fi and Bluetooth,” in *Proceedings of Ubiquitous 2014*, Dec. 2014, pp. 171–177.
- [20] T. Yoshida and Y. Taniguchi, “Estimating the number of people using existing WiFi access point based on support vector regression,” *Information*, vol. 19, no. 7A, pp. 2661–2668, Jul. 2016.
- [21] T. Mabuchi, Y. Taniguchi, and K. Shirahama, “Person recognition using Wi-Fi channel state information in an indoor environment,” in *Proceedings of IEEE ICCE-TW 2020*, Sep. 2020, pp. 567–568.
- [22] T. Xin, B. Guo, Z. Wang, M. Li, Z. Yu, and X. Zhou, “FreeSense: Indoor human identification with Wi-Fi signals,” in *Proceedings of IEEE GLOBECOM 2016*, Dec 2016, pp. 1–7.
- [23] 梶拓真, 大場春佳, 水野信也, “密集度検出システムの構築と運用,” *インターネットと運用技術シンポジウム論文集*, pp. 101–102, Dec. 2020.
- [24] 東京大学, “MOCHA,” <https://mocha.t.u-tokyo.ac.jp/>.
- [25] 徳田瑛, 荒川豊, 高野茂, 石田繁巳, “WiFi と BLE によるハイブリッド混雑度計測における自動パラメータ調整の検討,” *情報処理学会研究報告*, Mar. 2021.
- [26] 丸紅情報システムズ株式会社, “BlueBeacon Tag SE,” <https://www.marubeni-sys.com/iot/beacon/>.
- [27] I. Harvey, “bluepy - Python interface to Bluetooth LE on Linux,” <https://github.com/IanHarvey/bluepy>.
- [28] Apache, “rotatelog - Piped logging program to rotate Apache logs,” <https://httpd.apache.org/docs/2.4/programs/rotatelog.html>.