

Raspberry Pi 5 を用いた 高精度位置データ取得車載器の開発

永尾圭佑¹ 清水陸斗¹ 小野弘太郎² 福山侑弥² 津曲優斗²
内林俊洋³ 末吉智奈佐⁴ 安武芳紘¹ 稲永健太郎^{1, a)}

概要: 著者所属の研究室では地域公共交通の利便性向上を図るため、「地域公共交通運行管理支援システム」を開発・運用している。現在、本システムはタブレット端末を用いて車両位置データの取得を行っており、各端末内の GNSS モジュールの性能差や設置環境の影響により位置データ精度が安定しない課題があった。そこで、本研究は Raspberry Pi 5 を用いた車載器を開発し、タブレット端末の位置データとの精度比較を行った。本車載器は任意の Web サーバにデータを送信でき、GNSS モジュールの設置位置の変更やシステム改良が容易であるという特徴を持つ。走行軌跡類似度評価の結果、本車載器の類似度・タブレット端末の類似度はそれぞれ 86.4%・75.6%であり、10.8 ポイントの精度向上を確認した。本稿では、主に本車載器の位置データ精度の優位性について述べる。

キーワード: 情報提供・地図情報、位置情報サービス、その他

Development and Evaluation of a Raspberry Pi 5-Based In-Vehicle Unit for High-Precision GNSS Location Data Acquisition in Community Bus Operations

Keisuke Nagao¹ Rikuto Shimizu¹ Kotaro Ono²
Yuya Fukuyama² Yuto Tsumagari² Toshihiro Uchibayashi³
Chinasa Sueyoshi⁴ Yoshihiro Yasutake¹ Kentaro Inenaga^{1, a)}

Abstract: Our research group operates a regional public transportation management support system that uses tablet devices for real-time vehicle location tracking. However, location accuracy has been inconsistent due to variations in built-in GNSS module performance and device placement constraints. To address this, we developed a Raspberry Pi 5-based in-vehicle unit with an external USB GNSS module, capable of transmitting data to any web server and allowing flexible hardware and software modifications. Evaluation on an actual community bus route in Shingu Town, Fukuoka Prefecture, showed trajectory similarity of 86.4%, compared to 75.6% for conventional tablets—a 10.8 percentage point improvement—demonstrating its viability as a higher-precision alternative.

Keywords: Information Provision and Map Information, Location-Based Services, Other

1. はじめに

1.1 研究背景

日本国内では、少子高齢化と人口減少が年々進行しており、地方公共交通は運転手不足や利用客減少といった深刻な課題に直面している[1]。地方公共交通路線の統合・減便・廃止などにより、民間企業による地方公共交通の維持が困難になりつつある。一方で、高齢ドライバーの運転免許返納に伴う移動制約者の増加により、地方自治体が主体となって運行するコミュニティバスの需要は高まっている。

著者が所属する地域公共交通運行管理支援グループで

は、地域公共交通が抱える課題を ICT の活用により解決するため、「地域公共交通運行管理支援システム」の開発・実運用・管理を行っている（図 1）。本システムでは、バス車内に設置されたタブレット端末から取得した位置データを、著者所属の研究室内に設置されたサーバに送信している。しかし、各端末に内蔵された GNSS モジュールの性能差に依存した精度の位置データを利用するため、位置データ精度の信頼性が不安定であるという課題が明らかになっていった。具体的には、安定した精度の位置データを取得できない影響により、バスの現在位置が道路上から離れた位置に表示される不具合が生じていた（図 2）。

¹ 九州産業大学 理工学部 情報科学科
Department of Information Science,
Faculty of Science and Engineering, Kyushu Sangyo University.

² 九州産業大学大学院 情報科学研究科
Graduate School of Information Science, Kyushu Sangyo University.

³ 九州大学 情報基盤研究開発センター
Research Institute for Information Technology, Kyushu University.

⁴ 九州産業大学 産学共創・研究推進本部
Industry-Academic Co-innovation and Research Promotion Headquarters,
Kyushu Sangyo University.

a) inenaga@is.kyusan-u.ac.jp



図 1 地域公共交通運行管理支援システム

そこで本研究は、Raspberry Pi 5 を用いて高精度位置データ取得車載器を開発し、本システムの位置データに関する課題を解決することを目的とする。第一段階として、他社製バスロケーションシステムの調査を行い、本研究遂行の際の参考とした。第二段階として、Raspberry Pi 5 を用いて他社製サービスへの依存度を軽減した、Raspberry Pi 5 を用いた車載器の開発・実装・評価を行った。

1.2 地域公共交通運行管理支援システムの現状

1.2.1 同システムの位置データ送信方法

システムで利用される位置データは、車内に設置されているタブレット端末である Lenovo Tab B11 [2] から取得している。さらに、タブレット端末は車両前方中央部の運転席付近に設置されており、窓際から数十 cm 離れている(車両運行の安全面やシステムの機能面から、設置位置を変更できない)。

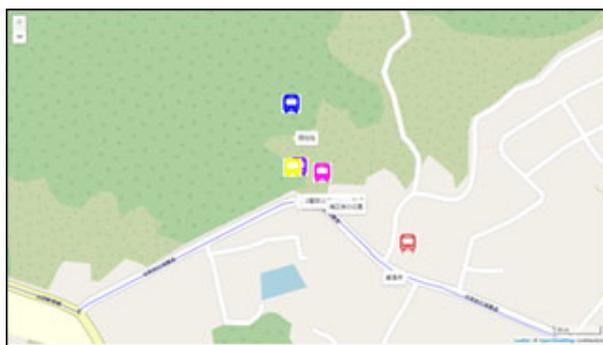


図 2 車両位置の表示不具合

また、位置データ送信のために 日本カバレッジ IoT SIM [3] を利用し、通信サービスは SORACOM Air for セルラー【plan-D (D-500MB)】 [4] を利用している。タブレット端末が取得した位置データは、本研究グループが運用するデータベースサーバ内に格納されている。

1.2.2 同システムが抱える不具合

同システムは車両の位置データを表示する際に不具合が発生するという課題を抱えている(図 2)。具体的には、車両が GNSS 測位精度の低下する地域または環境下を走行する際、取得された位置データと実際の運行ルートとの間に大きな誤差が発生する不具合が生じる場合がある。加えて、タブレット端末から取得する位置データ精度が不安定であることや、各端末の設置位置または内蔵された GNSS モジュールの性能に位置データの精度が依存することも問題となっている。

同システムのプログラムによる位置データ補正では、手動でのしきい値設定や車両現在位置の補正を行っている。これらの要素に加え、GNSS 衛星から受信できる信号の精度が低い等の理由で正確な位置データを得られにくい場合、補正後の位置データ精度にも限界がある。つまり、システムのプログラムによる技術的な位置データ補正処理だけでは限界がある。同システム上では、表示される位置データに数 m ~ 数十 m 単位の誤差が発生している現状がある。



図 3 いちごロケ (概要)
(いちごロケ公式ページより引用)

2. 関連研究

本章では、本研究遂行の際に参考とした、既存の他社製バスロケーションシステムについて述べる。ここでは、「いちごロケ」・「X-Loca」というサービスについて、それらの概要を述べる。

また、上記で挙げたシステムの他にも、本研究と類似または関連する研究内容も存在する[5][6]。

2.1 いちごロケ

いちごロケは、Sujiya Systems が開発したバスロケーションシステムであり[7]、これは IchigoJam 互換機・外付け GNSS モジュール・LTE 通信用モジュールを用いて安価に作成可能なバスロケーションシステムである(図 3) [8]。

位置データの通信には sakura.io のサービスを利用し、LTE 通信用モジュールを利用することにより各種データの送受信を行う。しかし、現在は関連機材の販売・製造が終了しているため、同一のシステム構築は事実上不可能である。

2.2 X-Loca

X-Loca は、株式会社ナチュラルスタイルが開発・提供するバスロケーション用 Web アプリケーションであり[9]、これは IchigoJam と MixSoda (通信機能追加用 IoT 拡張ボード)、GNSS モジュールを使用して製作できる車載器システムである(図 4) [10]。

また、位置データの通信に MixServer というサービスを利用し、LTE 通信用モジュールを利用することにより各種データの送受信を行う。

2.3 GTFS-JP

GTFS-JP は、バス運行事業者と利用者間で運行情報の受け渡しに活用される公共交通用データの共通フォーマットであり、17 のテキストファイルで構成される。Raspberry Pi 5 を用いた車載器は、任意のサーバに対して GTFS-JP の shapes.txt 内のデータ形式に準拠した位置データを送信す

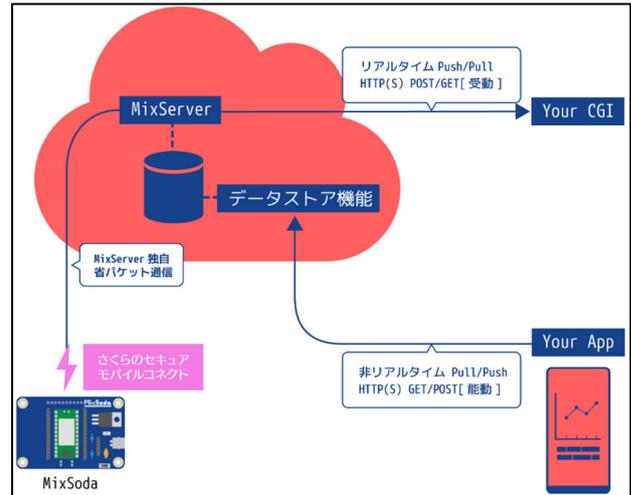


図 4 X-Loca (概要)
(X-Loca 公式ページより引用)

る。

本研究で述べる shapes データ (shapes.txt) とは、バス車両がどの地点を通るかを表すために必要となる緯度・経度情報を指し、車両が通過すべき地点の緯度・経度情報を含んでいる。つまり、この shapes データはバスが本来通過すべき地点である。本研究は shapes データ基準にすることで、位置データ精度の評価を行う。

3. 他社製バスロケーションシステムの問題点

本章では、前章で述べた他社製バスロケーションシステムの問題点について述べる。

いちごロケは、位置データの通信に sakura.io というサービスを利用するが、同プラットフォーム用の各種拡張キットの販売・製造を既に終了している[11][12]、そのため、同一機材を新たに準備することはできず、同システムの新規構築は不可能で加えて、sakura.io 経由での通信が前提であるため、任意のサーバに対して位置データを送信できない。X-Loca も同様に、MixServer という通信サービスを利用しているため、任意のサーバに対して位置データを送信できない。

以上のように、どちらのシステムも他社のサービスに依存する構造である。そのため、位置データ送信先サーバの変更が行えず、各種サービスの仕様・制限の影響を受け、柔軟なシステム改良が困難である。さらに、互換性のある機材の使用が求められる制約や、システム運用の際に使用できるサービスが限定される等の様々な課題も存在する。

4. Raspberry Pi 5 を用いた車載器の開発

4.1 Raspberry Pi 5 を用いた車載器の概要

他社製バスロケーションシステムの課題であった、任意のサーバに位置データを送信できない・柔軟なシステム改

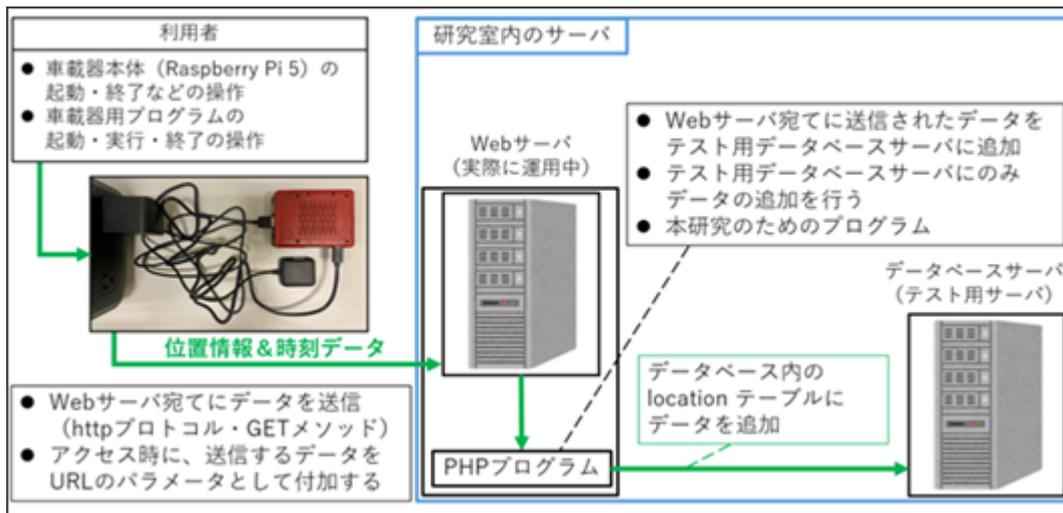


図 5 Raspberry Pi 5 を用いた車載器 (概要)

良や機能拡張に制約があるという問題を解決するために Raspberry Pi 5 を用いた車載器を開発した。Raspberry Pi 5 を用いた車載器は、GNSS モジュールの設置位置を車両窓際付近などの位置に自由に固定・移動でき、より安定した条件下での位置データ取得が可能である。ただし、Raspberry Pi 5 を用いた車載器では、Raspberry Pi 5 本体および車載器用プログラムの起動・実行・終了処理を手動で行う必要がある。以下の理由から、車載器開発の際に Raspberry Pi 5 を選定した。

- 少なくとも 2036 年 1 月まで生産が継続される見込みがあり、機材入手困難になるリスクが低いと判断したため。
- 専用 OS (Raspberry Pi OS) のインストールが可能で、UI ベースの操作やシステム管理の自動化など将来的な機能拡張に対応できると判断したため。

車載器用プログラムは、本研究グループの Web サーバ上で動作している PHP プログラムに対して http アクセス (GET メソッド使用) を行う。同 Web サーバ上で動作している PHP プログラムは、テスト用データベースサーバ内の location テーブルにデータの追加を行うためのプログラムである。つまり、Raspberry Pi 5 を用いた車載器から送信されるデータは、Web サーバ内で動作している PHP プログラムによってテスト用データベースサーバに追加される (図 5)。

4.2 Raspberry Pi 5 を用いた車載器のシステム設計

Raspberry Pi 5 を用いた車載器本体 (図 6) にはシングルボードコンピュータである Raspberry Pi 5 (図 6 の(C)) を採用し、外付け GNSS モジュール (GU-502MGG-USB) (図 6 の(A)) [13] と電源供給用の外部バッテリー (図 6 の(D)) を接続して車両の位置データを取得する構成をとる。なお、

外付け GNSS モジュールの位置データ出力形式は、「NMEA0183」である (表 1)。また、本車載器でプログラムを実行するためには、別途機材操作の入出力装置であるキーボード (図 6 の(E))・マウス (図 6 の(E))・モニター (図 6 の(B)) も併せて接続する必要がある。

車載器用プログラム (Python) は、GNSS モジュールが衛星から受信した信号データの取得・送信処理を行う。また、位置 (緯度・経度) 情報と送信時刻を、本研究グループの Web サーバに対して任意の間隔 (初期値: 3 秒) で送信する機能を持つ。これらのデータは、Web サーバ上の PHP プログラムを経由してテスト用データベースサーバに記録される。なお、車載器用プログラムは、無線 LAN (Wi-Fi) 環境下でのデータ送受信を前提として動作する。

4.3 Raspberry Pi 5 を用いた車載器のシステム実装

4.3.1 車載器システム実装の概要

Raspberry Pi 5 を用いた車載器のシステム実装は、Raspberry Pi OS のインストールおよび初期設定、車載器用プログラムの実行環境構築、車載器用プログラムの作成および実装の 3 段階で遂行した。システム実装時の各種作業工程・手順については、主に生成 AI 「Gemini」から提供された情報を参考にしている。

表 1 外付け GNSS モジュールの仕様

製品型番	GU-502MGG-USB
内蔵チップ	MediaTek 社製 MT3333
対応する GNSS	QZSS, GPS, GLONASS, SBAS, BeiDou, Galileo
対応する測位信号	L1C/A, L10F
受信チャンネル数	99
位置測位精度	2.5m CEP (-130 dBm)
更新周期	1~10Hz (標準: 1Hz)
位置データ出力形式	NMEA0183
動作電圧	4.5V~5.5V

4.3.2 車載器システム実装前の事前準備

Raspberry Pi OS のインストールおよび初期設定では、まず Raspberry Pi 5 の動作確認を実施した。OS プリインストール済みの microSD カード[14]を挿入して本体を起動したところ、正常に起動できない問題が発生した。その原因を調査した結果、同 microSD カード内のブートローダーファイルの一部が破損していたことが判明し、本体の LED インジケータランプ (ACT ライト) が 4 回点滅する現象も併せて確認された。これらの問題を解決するため、市販の microSD カード (OS インストール用 microSD カード) に対し、OS インストール用ソフトウェア「Raspberry Pi Imager」を使用して正常なブートローダーファイルをインストールする作業を実施した。ブートローダーのインストール完了後、Raspberry Pi 5 が正常に起動できることを確認し、続いて OS インストール用 microSD カードに対して Raspberry Pi OS をインストールした。OS 導入後は、ログインユーザーアカウントの新規作成、無線 LAN (Wi-Fi) との接続設定、システム内で使用する言語や時刻の設定等の初期設定作業を行った。一部のシステム設定項目は OS インストール前にも設定可能であり、デバイス再起動時にシステムに反映される。さらに、車載器用プログラム内で GNSS モジュールとの通信を行う必要があるため、Raspberry Pi 5 のシリアル通信 (UART) の有効化作業を実施した。この作業は、Raspberry Pi 5 のターミナルに「sudo raspi-config」と入力して表示される画面から、「3 Interface Options > I6 Serial Port」の項目を選択することで設定でき、設定反映のためには本体の再起動を要する。加えて、Raspberry Pi 5 本体の冷却用ファンの自動起動設定も行った。この設定を完了させるためには「サービスファイル (ファイル拡張子: service)」を

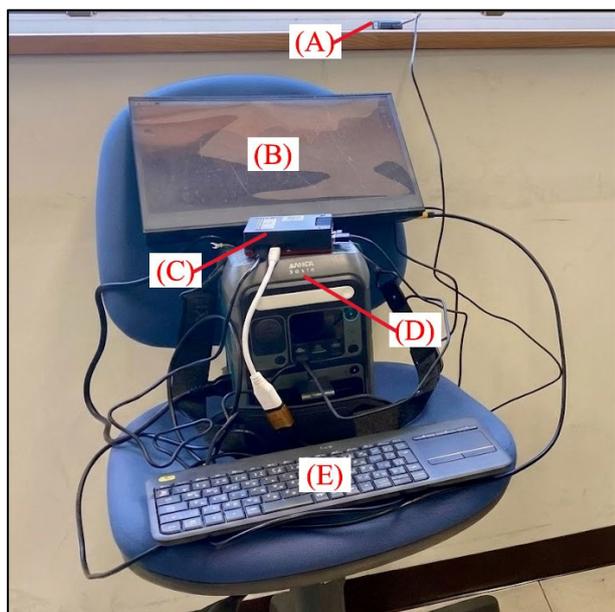


図 6 Raspberry Pi 5 を用いた車載器 (機材の全体図)

作成し、システム起動時に実行される処理を定義する必要がある。なお、機材の動作の安定性向上を目的として、この作業を行った。

4.3.3 車載器システムの実装

Raspberry Pi 5 上で車載器用プログラムを実行するために、車載器用プログラムを動作させるために必要となる Python ライブラリのインストール作業を実施した。車載器用プログラムは Python 言語で記述されており、GNSS モジュールからのシリアル通信データの取得 (serial ライブラリ・pynmea2 ライブラリ) や HTTP 通信 (request ライブラリ) による Web サーバへのデータ送信を行う。また、車載器用プログラムを OS パッケージマネージャの環境に依存することなく実行させるため、Python プログラム実行用の仮想環境を構築した[15]。つまり、車載器用プログラムは、この仮想環境上で各種処理を実行する。

車載器用プログラムの生成時には Gemini (生成 AI) を補助的に利用し、著者が動作検証および改良を行った。その後、統合開発環境「Visual Studio Code」を利用してプログラムの軽微な改良や動作確認を実施した。まず、GNSS モジュールの動作確認を主目的として、任意の Web サーバに対するテスト版データ送信用プログラムを作成した。これと併せて、テスト版データ送信用プログラムから送信されたデータを閲覧するためのローカル Web サーバ用プログラムも作成した。これらのプログラムは車載器の動作確認を目的として作成した。その後、車載器用プログラムの作成に着手した。車載器用プログラムに最終的に搭載された機能は以下の通りである。なお、以下の機能のうち、(4)(6)(7) は本車載器の動作確認用途で利用できる機能である。

- (1) Web サーバへのデータ送信、および送信されたデータ (ターミナル上に表示されたテキストデータ) の保存の可否を設定できる機能
- (2) シリアル通信により、GNSS モジュールから位置データを取得する機能
- (3) 任意の Web サーバ、または本研究グループの Web サーバ内の PHP プログラムに対して、位置および時刻データ送信を実行する機能
- (4) データ送信に関するステータス (送信成功、またはエラー情報) を画面上に表示する機能
- (5) NTP サーバと車載器本体との間で時刻情報を同期する機能
- (6) クラウドストレージ上のファイルや車載器のローカルストレージ上のファイルに対して、Web サーバに送信したデータを記録する機能
- (7) 車載器のローカルストレージ上のファイルに対して、車載器用プログラム実行時にターミナルに表示され

た内容を保存する機能

車載器用プログラムの動作確認には、PostgreSQL データベースサーバ管理用のオープンソースソフトウェアである PgAdmin4 を利用した。車載器用プログラムの実行時に、テスト用データベースサーバ内の location テーブルに格納されたデータの閲覧と、Raspberry Pi 5 のターミナル上に表示される情報の確認を行うことで、両者のデータの整合性を確認した。

5. Raspberry Pi 5 を用いた車載器の性能評価

5.1 性能評価の概要

福岡県糟屋郡新宮町のコミュニティバス「マリックス」の山らいず線（西鉄新宮駅～佐屋停留所間）に乗車して、実際の車両位置データを取得した。データ通信を行う際には、モバイル Wi-Fi ルーター（Aterm MR51FN）[16]を使用した。ただし、Raspberry Pi 5 を用いた車載器からは、佐屋方面行き乗車時の位置（緯度・経度）情報のみ正常送信でき、時刻情報は正常に送信できなかった。

車両位置データの取得後、バス車内のタブレット端末・Raspberry Pi 5 を用いた車載器から取得された位置データを、基準となる運行経路情報（図 7）と比較し、走行軌跡類似度を算出して精度を評価した。なお、基準となる運行経路情報とは、GTFS-JP の shapes.txt ファイル内の緯度・経度情報から描画した車両本来の走行経路であり、その生成には同コミュニティバスの shapes データを利用している。

5.2 性能評価の方法

まず、マリックスの山らいず線で運行されているバスに乗車し、バス走行中にタブレット端末・Raspberry Pi 5 を用

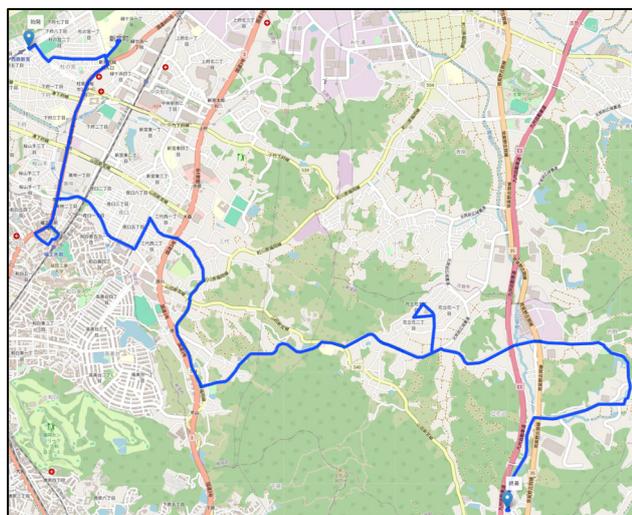


図 7 基準となる運行経路情報

いた車載器から同時に位置データを取得した。バス車内では、テスト用データベースサーバへの送信データを適宜確認した。

次に、取得された位置データから走行軌跡の類似度を算出した。本研究における「類似度」とは、実際に取得された位置データを表す走行軌跡が、shapes データが表す走行軌跡とどの程度一致しているかを数値化した値を指す。なお、車載器の性能評価は、正確な位置データを取得できた区間（佐屋方面行き）のみを対象として行い、基準となる運行経路情報（GTFS-JP データ）と比較対象となる各機器の位置データとの間の類似度を、著者が作成した類似度評価プログラムを用いて算出した。

類似度評価は、以下の手順で行った。

- (1) 使用する位置データを、類似度評価プログラムへ読み込む。
- (2) プログラム上で、正常に位置データを取得できている運行区間を指定する。
- (3) プログラムが算出した類似度を基に、各車載器の位置データ精度の安定性や優位性を評価する。

類似度評価プログラムは、以下の手順を実行することで類似度の値 (%) を算出する。

- (1) 比較対象の軌跡から取り出した一点と、基準軌跡を構成する全ての点との距離をそれぞれ計算する。
- (2) 計算した距離の中から、最も短い距離（最短距離）を一つだけ選出する。
- (3) 比較対象の軌跡に含まれる全ての点に、上記手順を繰り返して実行する。
- (4) 各点で見つけ出した最短距離を全て合計する。
- (5) 求めた最短距離の合計値を比較対象の点の数で割ることで、「平均距離」(=2 地点間の最短距離の平均値) を算出する。
- (6) 「類似度 (%) = 100 - 平均距離 (m)」の式に平均距離を代入し、類似度の値を算出する。

類似度評価プログラム内で 2 地点間の最短距離算出を行う際には、「ハバーサインの公式」を利用している。これにより、2 地点間の緯度・経度に加え、地球の曲率も考慮した最短距離を算出している。

本評価区間における平均距離 (shapes データとの誤差の程度を表す指標) は十数 m ～ 二十数 m 程度であり、この指標による有意な位置データ精度比較が可能である。なお、平均距離が 100m を超える状況では類似度 0% となるため、本プログラムの類似度算出機能は正常に動作しない。

5.3 性能評価の結果

類似度評価用プログラムにより算出された類似度は、タブレット端末が75.6% (図8)、Raspberry Pi 5を用いた車載器が86.4% (図9)であった。

上記の算出結果から、Raspberry Pi 5を用いた車載器の類似度がタブレット端末の類似度を10ポイント程度上回っていることが分かる。また、同車載器を用いた方が、より正規の運行ルートの走行軌跡を再現でき、精度の高い位置データを取得できていることが確認された。

6. 本研究の総括

本研究は、本研究グループが運用する地域公共交通運行管理支援システムにおける、タブレット端末内のGNSSモジュールにより取得された位置データ精度が不安定であるという課題に対し、ハードウェアに着目した解決を図るためにRaspberry Pi 5を用いた車載器を開発した。

まず、他社製バスロケーションシステムとの比較においては、機材面では大きな優位性が認められる。いちごロケはsakura.ioの各種拡張用キットの製造・販売終了により同システムの新規構築が不可能であり、X-Locaは位置データ送信先が特定のサーバに限定されるため他社サービスへの依存度が高いと言える。つまり、他社製バスロケーションサービスでは、システム構築の際に互換性のある機材を使用することやサービスの仕様により制限された範囲でのシステム構築を求められ、柔軟なシステム構築や改良が困難であることが分かった。

これに対し、Raspberry Pi 5を用いた車載器は任意のWebサーバに対してデータを送信でき、GNSSモジュールや機材の設置位置の変更も容易である。また、周辺機器接続用の各種ポートや拡張用機材接続用のピンヘッダを搭載しているため、外部機器や機能拡張用機材との接続が容易であ

り、利用者による柔軟なシステム改良にも対応可能である。加えて、Raspberry Pi シリーズは少なくとも2036年1月まで生産が継続されるため、将来的に機材を入手できなくなる可能性も低い。

位置データ精度に関しては、Raspberry Pi 5を用いた車載器の優位性は、既存の位置データ取得方法よりも高いことが示唆された。走行軌跡の類似度評価により双方の位置データ精度を比較した結果、Raspberry Pi 5を用いた車載器の類似度の値はタブレット端末の類似度の値を上回っていた。したがって、本システム評価の範囲内では、Raspberry Pi 5を用いた車載器がタブレット端末よりも高精度な位置データを取得できることが確認された。

さらに、Raspberry Pi 5を用いた車載器は位置データの活用や外部システムとの連携を行う場面、または機材の設置や整備に関する制約が少ない場面での運用に適していると言える。

これらを総括すると、Raspberry Pi 5を用いた車載器は、本研究グループで運用されているシステムの位置データ取得方法の代替手段として利用できる可能性が高いと言える。

7. 今後の課題

Raspberry Pi 5を用いた車載器全体の問題として、バス車内に持ち込む機材が多く運搬や動作確認に時間を要する点が挙げられる。また、現状では車載器用プログラムを完全手動操作で実行しているため実用性・利便性が高いとは言えず、プログラムの自動起動・実行・終了等のシステム全体の自動化を実現する必要がある。さらに、車載器の設置位置・管理方法や動作電源の安定的な確保手段の検討に加え、バスの運行事業者や運転手に対する作業負担を可能な範囲で軽減できるシステム構成の実現も求められる。

また、車載器用プログラムにおいて、位置データに付加

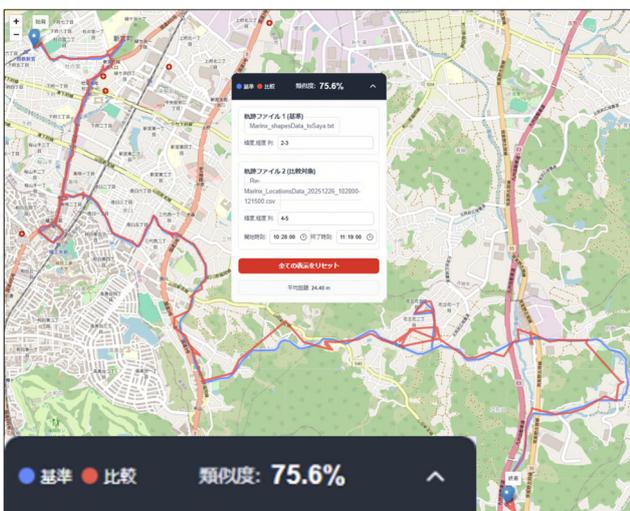


図8 走行軌跡の類似度 (車両内のタブレット端末)

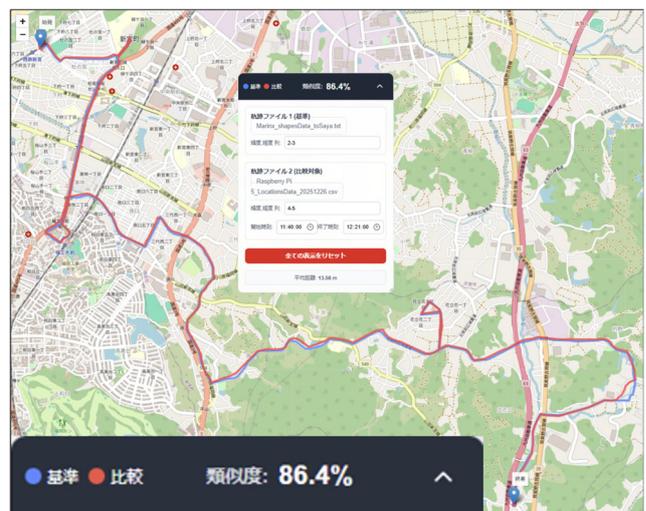


図9 走行軌跡の類似度 (Raspberry Pi 5を用いた車載器)

されている時刻情報が Web サーバに正常に送信できない不具合があり、これは優先的に対処すべき課題であると言える。この不具合は、プログラムの待機処理中に GNSS モジュールから取得されたデータが送信待ちバッファに蓄積し、データ送信処理に遅延が発生していることが原因と推測される。そのため、搭載した GNSS モジュールの仕様に適した車載器プログラムの改良も必要になると考える。

謝辞

本研究を行うにあたり、開発したシステムの評価に関する作業に快く協力して下さった、福岡県糟屋郡新宮町役場 産業振興課の皆様、新宮タクシー株式会社コミュニティバス事業本部 関係者の皆様に対し、深く感謝申し上げます。

参考文献

- [1] 国土交通省：令和7年度第1回（第24回）交通政策審議会交通体系分科会地域公共交通部会 配布資料 【資料1-1】地域公共交通の現状，入手先
(<https://www.mlit.go.jp/policy/shingikai/content/001898150.pdf>) (参照 2026-02-11)。
- [2] Lenovo：Lenovo Tab B11 | エンターテインメント性能に優れた大画面タブレット | レノボ・ジャパン，入手先
(https://www.lenovo.com/jp/ja/p/tablets/lenovo-tab-series/lenovo-tab-m11-11-inch-mtk/len10310022?ipromoID=Megamenu_tablet_3_Lenovo-Tab-B11-11-inch) (参照 2026-02-11)。
- [3] 株式会社ソラコム：日本カバレッジ IoT SIM，入手先
(https://soracom.jp/services/air/japan_coverage/) (参照 2026-02-11)。
- [4] 株式会社ソラコム：SORACOM Air for セルラーの特徴，入手先
(<https://users.soracom.io/ja-jp/docs/air/feature/>) (参照 2026-02-11)。
- [5] 遠藤雅樹，品川達郎，山中光定，人見功治郎，高尾和志，大野成義，石川博：地域公共交通に適応したバスロケーションシステムの開発，情報処理学会論文誌(トランザクション)，Vol. 7, No. 2, pp. 117-134, 2014
- [6] 米本拓馬，西尾航，鈴木秀和，松本幸正：GTFSを用いたリアルタイムバス停ナビゲーションシステムの提案，情報処理学会全国大会，Vol. 2017, No. 1, pp. 407-408, 2017
- [7] Sujiya Systems：いちごロケ，入手先
(<https://www.sinjidai.com/ichigo/>) (参照 2026-02-09)。
- [8] さくらインターネット株式会社：さくらインターネットが提供する IoT プラットフォームサービス，sakura.io，入手先
(<https://sakura.io/>) (参照 2026-02-09)。
- [9] 株式会社ナチュラルスタイル - NEWS (PR 資料)：DIY なら1車両の通信費が月額 330 円！ 自社でつくるバスロケーションシステムを応援します！，入手先
(<https://na-s.jp/PR/PR231228.pdf>) (参照 2026-02-09)。
- [10] 株式会社ナチュラルスタイル：MixServer ~ a IoT device & service，入手先
(<https://mixsoda.io/>) (参照 2026-02-09)。
- [11] さくらインターネット株式会社：「sakura.io LTE モジュール」製造終了のお知らせ，入手先
(<https://www.sakura.ad.jp/corporate/information/announcements/2021/04/08/1968206631/>) (参照 2026-02-09)。
- [12] さくらインターネット株式会社：【sakura.io】供給元の部品取り扱い中止に伴うオプション品製造終了のお知らせ，入手先
(<https://www.sakura.ad.jp/corporate/information/announcements/2022/07/29/1968210010/>) (参照 2026-02-09)。
- [13] 株式会社秋月電子通商：[117981]GPS 受信機 USB 接続タイプ みちびき 2 機(194/195)対応 GU-502MGG-USB，入手先
(<https://akizukidenshi.com/catalog/g/g117981/>) (参照 2026-02-09)。
- [14] 株式会社スイッチサイエンス：Raspberry Pi A2 クラス microSD カード (64GB, OS プリインストール済) --在庫限り，入手先
(<https://www.switch-science.com/products/10007>) (参照 2026-02-09)。
- [15] Python documentation：venv --- 仮想環境の作成，入手先
(<https://docs.python.org/ja/3/library/venv.html>) (参照 2026-02-09)。
- [16] NEC (日本電気株式会社)：Aterm MR51FN | 製品一覧，入手先
(<https://www.aterm.jp/product/atermstation/product/mobile/mr51fn/>) (参照 2026-02-11)。