

スマートフォンを用いた車線推定手法

牧野友哉[†] 伊藤嘉博[†] 中村嘉隆^{††} 高橋 修^{††} 白石 陽^{††}

近年, 携帯端末の中でも多種のセンサを内蔵したスマートフォンの普及が進んでいる. そして, そのような携帯端末を利用して低コストかつ広域にわたるセンサデータを収集する参加型センシングが注目されている. 一方, 道路交通における安全化・効率化システムの中でもプローブ情報システムが注目されており, 現在 VICS (Vehicle Information and Communication System) などのシステムが国内で導入されている. しかし, 既存の道路交通システムでは路上ビーコンや通信基地局の設置, また, それを利用する専用カーナビの導入が必要であり, システムの導入コストの低減や普及率の向上が課題となる. そこで, 本稿ではスマートフォンを利用した参加型センシングによる走行情報の収集と, 走行車線の推定への活用法を提案する. 走行車線情報は交通の安全化・効率化システムにおいて実用可能な情報であり, その推定技術は現在盛んに研究が行われている. 提案した車線推定システムを設計し, 提案システムの有効性について検証する.

A method for traffic lane estimation with a smartphone

TOMOYA MAKINO[†] YOSHIHIRO ITO[†]
YOSHITAKA NAKAMURA^{††} OSAMU TAKAHASHI^{††}
YOH SHIRAISHI^{††}

In recent years, smartphones with a variety of sensors become widespread. Participatory sensing that collects sensor data over a wide area with low cost using mobile devices has been attracting attention. On the other hand, Probe information systems have been attracting attention to realize safety and efficiency of transport system. And now, systems such as VICS (Vehicle Information and Communication System) have been introduced. However, in the existing systems, reduction of the introduction cost and the increase of the penetration rate will be issues because communication base stations, optical beacons, and the specific car navigation system are required for the systems. In this paper, we propose a method to collect car travel information with participatory sensing by a smartphone and to estimate the car traffic lane. Traffic lane information is important and available to realize safe and efficient traffic systems. Finally, we implemented the proposed system and evaluate the effectiveness of the system.

[†]公立はこだて未来大学大学院 システム情報科学研究科
Graduate School of Systems Information Science, Future University Hakodate

^{††}公立はこだて未来大学大学 システム情報科学部
School of Systems Information Science, Future University Hakodate

1. はじめに

道路交通の安全化・効率化を目的としたシステムとして, プローブ情報システムが現在注目されている[1-3]. プローブ情報システムは自動車に内蔵されているセンサから得られる位置情報や外気温, エンジン回転数, ワイパーなどのプローブ情報を収集し, ナビゲーションや安全運転支援などに活用するシステムである. プローブ情報は粒度が高く多種多様なセンサデータであるため, リアルタイム性の高いデータだけでなく蓄積したデータも有効に活用可能であり, 多くの活用例が挙げられている[4,5]. プローブ情報システムを実用化した例として VICS (Vehicle Information and Communication System) が有名である[6,7]. VICS の特徴として, 走行中に収集したプローブ情報を専用カーナビから路上ビーコン, FM 多重放送基地局を介してデータベースに収集し, 加工した情報を再びカーナビ向けに配信することで渋滞ナビゲーションや天候情報提供, エコドライブ支援などを実現している.

しかし, VICS のような従来のプローブ情報システムを利用するためには専用機器の導入や路上ビーコン, 通信基地局の設置が必要であるため, そのコストの削減に向けた代替法の提案が課題となる. また, プローブ情報の蓄積後の利用方法として, 安全化・効率化へ活用される例はあまりない. それは, 安全化・効率化のシステムにおいて情報のリアルタイム性が重視されるためであると考えられる. そのため, プローブ情報がこのような用途で活用されるためには, 粒度の高い情報を低コストで収集・蓄積する必要がある.

一方, 近年 MEMS (Micro Electro Mechanical Systems) の発展によるセンサデバイスの小型化, 高性能化に伴い, 私たちの身の周りのあらゆるものにセンサが組み込まれている[8,9]. それに伴い, スマートフォンを持ち歩く歩行者や自動車, Web データなどをセンサとみなし, 広範囲なセンサ情報を収集するセンシングを参加型センシング (Participatory Sensing) 技術の研究が進んでいる[10,11]. 参加型センシングはセンサとみなす媒体数が多く, またその種類も豊富であるため, プローブ情報よりもさらに広域かつ多種多様な情報を容易に収集できる. このような特徴から, 都市レベルでのセンシングも可能であり, 様々な事象を解析・理解するアプローチとして現在注目を浴びている. つまり, 参加型センシングにより収集した情報を交通の安全化・効率化に活用することで, 従来のプローブ情報システムの問題を解決することができると思われる.

そこで, 本稿では現在普及率の高いスマートフォンに内蔵されているセンサを利用することで, プローブ情報を低コストで収集・蓄積するとともに, その蓄積した情報を走行車線の推定に活用する手法を提案する. 交通の安全化・効率化のシステムの基盤となる技術の中でも車両の走行車線の推定は重要視されており, 事故の状況分析や車線ごとのナビゲーションに役立てることが可能である. これらのことか

ら、スマートフォン内蔵のセンサにより収集したログデータを分析し、各地点で走行していた車線を推定するアルゴリズムの実装および推定結果の精度評価を行うことを本研究の目的とする。

2. 関連研究

本章では、交通システムにおける安全化・効率化の基盤技術となる車線推定の関連技術、関連研究について述べる。

2.1 GPS による位置推定

GPS (Global Positioning System) は地球上を周回する人工衛星を利用して自分の位置を測位するシステムである。また、位置推定精度向上のために様々な技術の導入が進んでおり、小型化、低コスト化による普及率の向上から、GPS は我々が持っている携帯電話の多くに組み込まれており、現在身近な存在である。本節では、GPS の既存技術として移動体端末に広く用いられている GPS の性能について説明する。

一般的な GPS は、地球上を周回する約 24 個の GPS 衛星から電波を受信し、位置情報を取得するシステムである。GPS の欠点として、電波を受信できる GPS 衛星の数が少なくなってしまう場合には位置精度誤差が大きくなってしまう。室内時や電波受信中に受信機器が建物の影に隠れてしまい十分な数の GPS 衛星から電波を受信できなくなってしまうと、最大で 100m の誤差が発生してしまうとされている。このような従来の GPS の問題を解決するために、近年携帯電話や自動車、船舶などには DGPS (Differential-GPS) や AGPS (Assisted-GPS) のような精度誤差を修正する技術が用いられている。

DGPS は GPS の位置情報取得の性能に加えて位置のわかっている基地局が発信する FM 放送の電波からも位置情報を取得することで、GPS 単体よりも高精度な位置推定を実現するシステムである[12,13]。位置情報の受信が可能な FM 基地局は日本全国に約 30 局あり、電波が受信可能な範囲であればどこでも DGPS の機能が利用可能である。通常の GPS では約 100m の誤差が生じるが、DGPS では約 5m の誤差に抑えることが可能になる。しかし、DGPS を利用するには、DGPS 対応型 GPS 受信機であり、一般的な GPS に比べて高価であるため、導入コストがかかるという問題がある。また、FM 放送を利用していることから、ノイズが混入してしまった場合は基地局から位置情報を取得できなくなってしまう。そのため、システムのロバスト性の向上も課題となる。

AGPS は DGPS と同様に通常の GPS の位置情報に補正を加えて更に高精度な位置推定を実現するシステムである[13,14]。DGPS との相違点として、AGPS では携帯電話基地局が衛星から位置情報を受信し、その基地局から携帯電話網を通じて位置情

報を取得する。AGPS は主に携帯電話に組み込まれており、携帯電話向けの地図アプリやナビゲーションなどのサービスに利用されている。また、FM 基地局に比べて携帯電話基地局は大量に設置されており、比較的ノイズに影響されにくいいため、AGPS の方が劣通信環境でも位置情報の精度が高いという特徴がある。しかし、AGPS の仕組み上、携帯電話サービスエリア外（圏外）では位置情報の取得は不可能であり、位置情報の測位の際には通信料が発生してしまう。また、GPS 衛星からの電波の受信と携帯電話基地局からの電波の受信を同時に行う必要があるため、通常の GPS に比べてバッテリーの消費量の増大が問題点として挙げられる。

2.2 車載カメラを用いた位置推定

既存の GPS では十分な精度が得られないため、車載カメラを利用して走行中の車両の位置を高精度に推定する研究がある[15]。この研究では、M-CubITS (M-sequence Multimodal Markers for ITS) というシステムを応用した手法を提案している。M-CubITS は画像のトラッキングを利用して走行している車線の領域や歩行している歩道の認識をしている。文献[15]ではこの M-CubITS を応用して走行中に車線の境界線となる白線をトラッキングし続け、白線をまたいだ際に車線変更を検知することで走行している車線の推定を実現している。あらかじめデータベースに各道路の車線数を格納しておき、走行中に位置情報をデータベースに問い合わせることで走行している道路の車線数を取得する。車線数が片側 2 車線以上の場合に隣接している領域は道路であるのか、道路以外（ガードレールや中央分離帯など）であるのかを認識することで自分が現在走行している車線の推定が可能になる。

2.3 ログデータを用いた位置推定

既存の位置推定手法では車載カメラやレーダーなどの利用価値の高いセンサデータを使用している反面、システムの導入コストが高い。一方、現在センサ機器の小型化、高性能化に伴い安価で利用価値の高いセンサデータの収集が可能になり、収集したログデータの活用が低コスト化に向けて注目されている。

文献[16]では、走行中に収集した車両の速度ログデータのみを用いて走行軌跡の推定を行っている。走行軌跡の推定手順として、まず、教師データ作成のための走行実験を事前に行い、大量の走行ログデータを収集する。収集したログデータの中でも車線変更の際の車両の動きを準備、移動、調節の 3 段階に対応付けした教師データを生成する。そして、走行軌跡推定に向けた本実験を行い、再度速度ログデータを収集する。収集した速度ログデータとあらかじめ用意した教師データを比較することで車線変更の際の 3 段階の動きを検知し、走行軌跡を推定している。更に、推定結果を状態推定アルゴリズムである隠れマルコフモデルを利用し、学習処理させることで推定精度の向上を図っている。そして、実環境において車線変更が不可

能である地点や認知的に車線変更を行わないと考えられる地点を登録した認知的ハ
いる。以上の手順を踏み、走行時の車両の移動軌跡を確率的に求めている。

文献[17]では、自転車に設置した Android 端末の内蔵センサ（加速度センサ、方位
センサ、GPS）を利用することで走行中の情報を収集・共有する機構 sBike を提案し
ている。この手法で推定する自転車の状態は、停止、直進、右折、左折、蛇行の 5
状態であり、まず、収集した走行ログデータを用いて各状態におけるセンサ値の変
化を割り当て、学習データを作成する。そして、本走行実験で収集したログデータ
の走行状態を隠れマルコフモデルによる学習処理を施すことで走行状態を推定する。
実験結果より、時間窓の幅や多数決アルゴリズムの適応は走行状態の継続時間に影
響を受けるため、継続時間によってそれぞれのパラメータを調整することで更なる
推定精度の向上につながるという考察が得られている。

2.4 まとめ

本章で紹介した関連研究に関して、各手法の利点・欠点をまとめたものを表 1 に
示す。

表 1 各手法の利点・欠点

手法		車線推定 (推定精度)	計算コスト	ロバスト性	導入コスト
GPS を用いた 位置推定	GPS	×	○	×	○
	DGPS[12,13]	×	○	×	○
	AGPS[13,14]	×	○	×	○
車載カメラを用いた 位置推定[15]		○	○	×	×
確率的手法を用いた 走行軌跡推定[16]		○	×	○	○
自転車走行状態推定[17]		×	○	○	○

○：可（目標値）

×：不可（改善の必要あり）

本研究の目的である車線推定において、推定精度の観点では[15,16]が有効だが、
計算コストとロバスト性、またシステムの普及に向けた導入コストの面で問題がある。
計算コストおよび導入コストの削減には GPS が有効であるが、位置推定の精度

ザードマップと推定結果を照合することで、誤推定を検知し推定精度を向上させて
が不十分であるため、車線推定の実現が困難である。スマートフォン内蔵の GPS に
より、収集した走行時の位置情報のログデータを図 1、図 2 に示す。

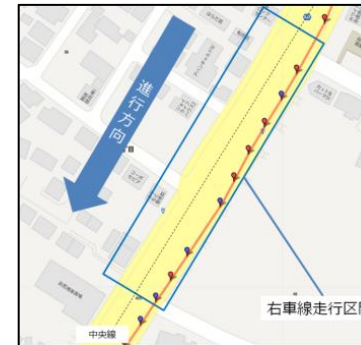


図 1 右車線走行時の軌跡



図 2 左車線走行時の軌跡

図 1、図 2 はそれぞれ、片側 2 車線道路の右車線と左車線を走行した際の GPS の
位置情報である。図 1 の左車線走行時の軌跡を見てみると、歩道に近い場所を通過
しているログが残っている。しかし、図 2 の右車線走行時の軌跡でも同様に歩道に
近い点にピンが打たれている。この結果から、GPS の補測精度では車線推定は困難
であることがわかる。[17]も GPS 同様、精度の観点では不十分であり、車線推定へ
の応用がなされていないが、スマートフォン内蔵のセンサにより収集したログデー
タのみを利用しているため、計算コスト・ロバスト性・導入コストの面では有効な
アプローチと言える。

3. 提案システム

本章では、既存研究の課題に対するアプローチと研究目標を述べ、提案手法のシ
ステム構成とその詳細について述べる。

3.1 アプローチと研究目的

既存研究の課題として、カメラを用いた手法では、計算コスト・ロバスト性・導
入コストの面が問題であった。また、GPS およびログデータを用いた手法では、車
線推定における精度が不十分であった。これらの問題を解決し、走行車線の推定を
実現するには低コストで有用性の高い走行ログデータを収集し、低計算コストで車
線推定に活用する方法が必要である。

そこで、本研究では現在普及率が高く、小型化・高性能化が進んでいるスマートフォン内蔵センサを利用して走行ログデータを収集するとともに、その収集したログデータのみを用いた走行車線の推定手法を提案する。現在、スマートフォンには様々な種類のセンサが組み込まれており、このセンサを利用して収集したログデータは非常に実用性が高いとされている。また、スマートフォンにはナビゲーションアプリやオーディオプレイヤーなどの機能も備わっており、運転中に車内のダッシュボードなどに設置して利用されるケースが多い。そのため、走行中にスマートフォンのセンサを起動して、センサログデータを収集した際のドライバーに与える手間や集中の妨げなどの負荷はさほど大きくないと考えられ、その収集したセンサログデータは走行中の車両の動きを推定するための有用な情報となる。また、スマートフォンのセンサはカメラやGPSなどに比べて車体に設置するだけで天候や時間帯などの外的環境に左右されないロバスト性の高いセンサデータが取得可能である。

以上より、スマートフォンのセンサを用いることで低導入コストかつロバスト性の高いセンサデータが容易に収集可能である。本研究ではスマートフォンを用いて収集したセンサログデータを低計算コストかつ高精度に車線推定に活用することを研究目的とする。

3.2 システム構成

提案システムの構成を図3に示す。本システムは、ログデータ収集システムと車線推定システムから構成される。ログデータ収集システムは、スマートフォンに内蔵されている加速度センサ、方位センサおよびGPSを利用し、車内にスマートフォンを設置して走行することで走行中の車両の加速度、車体の向き、位置情報（緯度、経度）を収集する。車線推定システムは、収集した走行ログデータから右左折した地点と車線変更を行った地点をそれぞれ検知し、その結果をもとに走行した車線を推定する。

車線推定システムは、次の手順でログデータの収集と車線推定を行う。

- ① 走行ログデータ収集
- ② 走行ログデータをデータベースに登録
- ③ 道路ネットワークデータの分割・登録
- ④ 右左折・車線変更の検知
- ⑤ 検知結果をデータベースに登録
- ⑥ 検知結果を利用し、全区間の車線を推定

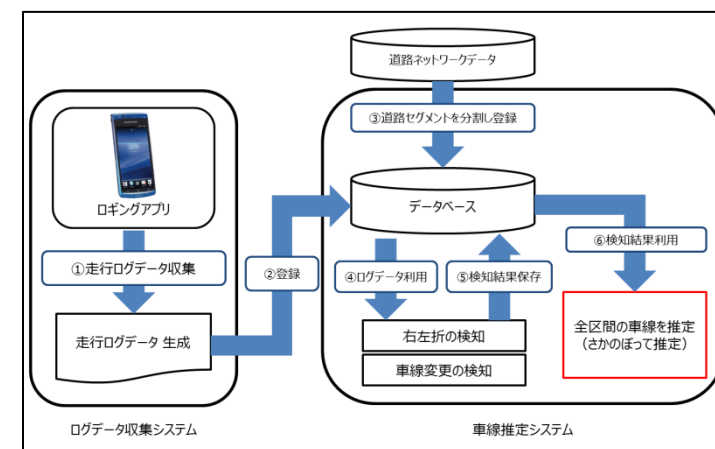


図3 システム構成図

まず、スマートフォンを車内に設置してログギンアプリケーションを起動しながら走行することで走行ログデータを収集する(①)。収集した走行ログデータは一旦端末内に保持し、走行後にデータベースに登録する(②)。車線推定の前準備として、道路ネットワークの各セグメントを分割したテーブルを生成し、データベースに登録する(③)。そして、走行ログデータを利用し、右左折と車線変更を検知し、検知した地点を分割したセグメントの属性データとしてデータベースに再度保存する(④、⑤)。最後に、⑤の検知結果を利用し、全区間の車線を推定する。提案システムの中でも重要となる車線推定の処理の流れを図4に示し、各手順について詳細に説明する。

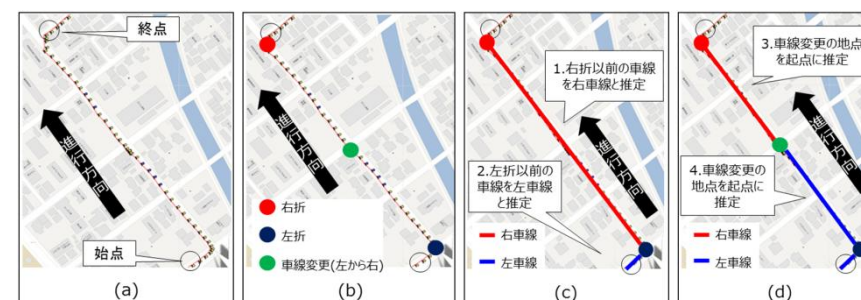


図4 車線推定処理の手順

車線推定は次の手順で行う（図 4）。

- (a) 始点と終点の認識
- (b) 右左折と車線変更の検知
- (c) 右左折の検知結果を車線推定に利用
- (d) 車線変更の検知結果を車線推定に利用

まず、収集したログデータのタイムスタンプから走行した際の始点と終点を検知し、それぞれの位置情報をデータベースに保存する(a)。次に、収集したセンサのログデータから右左折と車線変更を行った地点を検知し、それぞれの位置情報をデータベースに保存する(b)。最後に、(a)、(b)の検知結果を利用して走行した全区間の車線を推定する。全区間の車線の推定にあたって、まずログデータを終点からスキャンしてゆき、右左折の結果のみを反映する(c)。そして、再びログデータを終点からスキャンし、車線変更の地点を反映し、全区間の車線推定を完了する(d)。各手順の詳細については以下で説明する。

3.3 車線推定

車線推定システムにおけるデータベーステーブルと処理手順について図5に示す。車線推定システムは道路ネットワークデータ、収集ログデータおよび推定結果を格納するためのデータベースを用いて構築する。道路ネットワークデータには実際の道路の交差点と湾曲している部分に点を打ち、それらの点を結んだ道路セグメント群が格納されている。つまり、実際の道路を全て直線で表現した道路データである。そのため、各セグメントは異なる長さで格納されている。

本節では、3.2 節で述べた車線推定システムの中でも最も重要となる右左折の検知、車線変更の検知、および検知結果を利用した全区間の車線推定について説明する。また、具体的な車線推定処理手順については 3.5 節で述べる。

3.3.1 右左折の検知

右左折はスマートフォン内蔵のセンサにより収集した方位ログデータを利用して検知する。方位センサは端末の向きに応じて 0 度~359 度の方角が取得できる。ログデータ収集の際、スマートフォンは車内に固定されているため、右左折した場合にはこの方位センサの値が大きく変化すると考えられる。方位ログデータの変化を検出することで右左折した地点を検知し、検知結果をデータベースに保存する。

そして、右左折の検知結果を利用し、右左折の直前に走行していた一定区間の走行車線を推定する。右左折検知結果の車線推定への利用方法については 3.3.3 項で後述する。

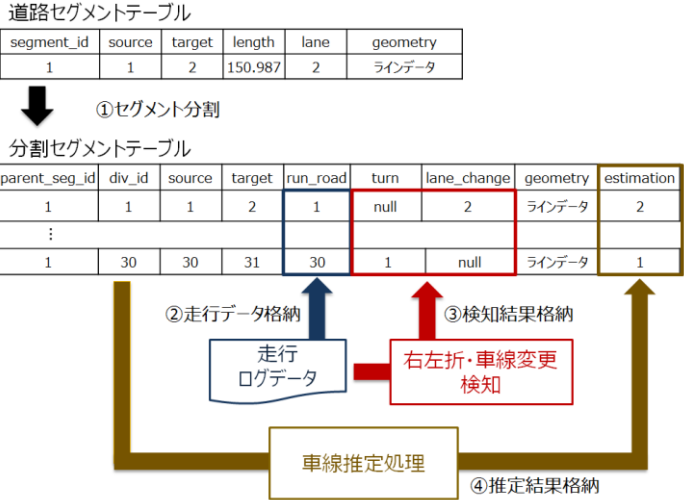


図 5 車線推定システムにおけるデータベーステーブルおよび車線推定処理手順

3.3.2 車線変更の検知

車線変更は、方位ログデータと道路の属性データが登録されている道路ネットワークデータに登録されている方位の値を比較することで検知する。道路ネットワークデータは実際の道路の中でも直線に近い区画をセグメント単位でテーブルに保存しており、属性として、各セグメントの ID、始点・終点の位置情報、中点の位置情報、長さ、方位および車線数が登録されている。

まず、道路ネットワークデータの方位の値とログデータの方位の値と時系列で比較してゆき、その差異を求める。そして、車線変更の閾値を設定し、差異の大きさが閾値以上であれば車線変更を行ったと判断する。検知結果は各道路セグメントを 5m ごとに分割した分割道路セグメントテーブル内に格納する。道路セグメントの分割幅が 5m である理由として、一般的な自動車の車体長は 5m 以下であること[18]、また、時速 60km で走行した際の 1 秒あたりに進む距離は約 16.7m であるため、車線推定結果の活用を考えた場合、5m 間隔での推定結果があれば十分活用可能であると考えられるためである。車線変更検知結果の車線推定への利用方法について 3.3.3 項に記述する。

3.3.3 全区間の車線推定

全区間の推定方法として、走行した経路を終点からさかのぼって右左折または車線変更の地点を検索し、その地点前後の車線を推定する。本稿では車線数を片側2車線と片側1車線の2つに限定し、右左折は交差点のみで行われるものとする。交差点の形状は次の3種類を想定する。

- 2-2 交差点
片側2車線と片側2車線の交差点
- 2-1 交差点（1-2 交差点）
片側2車線と片側1車線の交差点（交差点進入時に走行している車線数が2車線の場合は2-1 交差点、1車線の場合は1-2 交差点）
- 1-1 交差点
片側1車線の交差点と片側1車線の交差点

これらの交差点形状に加えて、交差点に進入する際の自車両の車線を考慮した場合、2-2 交差点では2つ、2-1 交差点では2つ、1-2 交差点では1つとなり、全部で6つの条件が考えられる。例として、図6の(a)に2-2 交差点進入の際の条件を示す。

2-2 交差点の場合、まず進入する際の車線が右車線と左車線の2つの条件に分けられる。そして、各条件での交差点進入後の動向が限定される。例えば、進入の際の車線が右車線であればその状態から左折することはないため、直進か右折の2択になる。同様に進入の際の車線が左車線であれば、直進か左折の2択に限定される。このように、交差点の種類と進入の際の車線が決まると侵入の際の車両の動向も限定される。この性質を考慮することで、右左折以前の車線が推定できる。つまり、右折以前に走行していた道路が片側2車線道路であった場合、その車線は右車線と推定し、左折以前は左車線と推定できる。

次に、車線変更の際の条件として、車線変更を行うのは直進道路に限定される。直進道路とは右左折が不可能であり直進することしかできない道路を指す（湾曲している道路も含む）。そして、直進道路を走行している際の車線によって次の車両の動向が限定される。直進道路の車両の条件として、片側2車線の場合の右車線および左車線走行時と片側1車線の場合の3条件が考えられる。片側2車線道路の左車線を走行していた際の車両の動向を図6の(b)に示す。

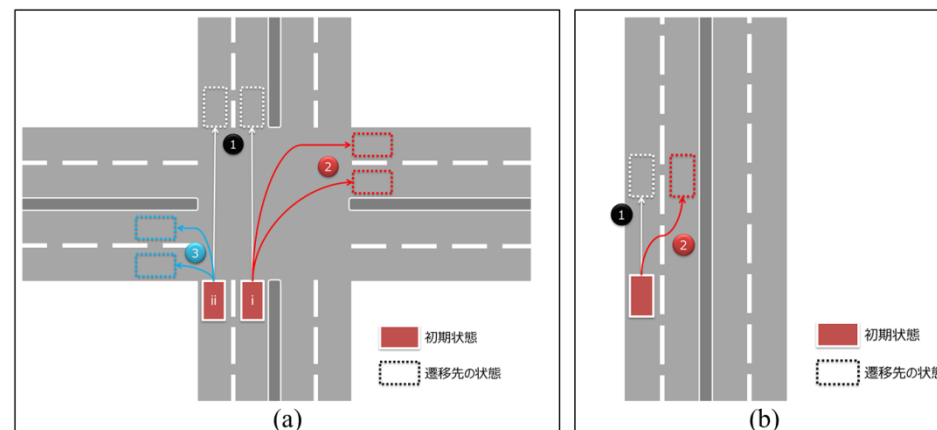


図6 道路状況別の車両の動向 (a: 2-2 交差点 b: 片側2車線道路)

- 2-2 交差点
 - i. 進入車線：右車線
 - ① 直進（右車線→右車線）
 - ② 右折（右車線→右車線，左車線）
 - ii. 進入車線：左車線
 - ① 直進（左車線→左車線）
 - ③ 左折（左車線→右車線，左車線）

- 片側2車線道路
 - i. 走行車線：左車線
 - ① 直進（左車線→左車線）
 - ② 左車線へ車線変更（左車線→右車線）

※括弧内は交差点進入前と進入後それぞれの車線の変化を示している

直進道路での走行車線が左車線の状態から右左折、および左車線へ車線変更は不可能であるため、車両の動向は直進か右車線へ車線変更の2択に限定される。同様に他の2つの条件の場合の車両の動向も限定される。そして、車線変更が検知できた際にはその地点を起点として車線変更前と車線変更後の走行車線が推定できる。例えば、図6(b)の②の車線変更が検知できた場合、検知前は左車線、検知後は右車線と推定できる。以上の交差点および直進道路での車両の動向について、表2にまとめる。なお、表中の「初期状態」は車両の動作を起こす前の車線であり、「車

線無し」は片側 1 車線道路を走行中の車線を指す。またハイフンは車両の動向が定義されない場合を指す。

表 2 車両の動向パターン

車両の動作および動作後の状態			直進	右折	左折	車線変更
環境		初期状態				
交差点	2車線-2車線	右車線	右車線	右 or 左	-	-
		左車線	左車線	-	右 or 左	-
	2車線-1車線	右車線	車線無し	車線無し	-	-
		左車線	車線無し	-	車線無し	-
	1車線-2車線	車線無し	車線無し	右 or 左	右 or 左	-
	1車線-1車線	車線無し	車線無し	車線無し	車線無し	-
直進道路	2車線	右車線	右車線	-	-	左車線
		左車線	左車線	-	-	右車線
	1車線	車線無し	車線無し	-	-	-

上記の条件分けにより、右左折および車線変更の検知結果から走行車線が推定できる。推定の手順としては 3.2 節の図 4 に示した通り、ログデータを終点から順に検索してゆき、まず右左折の検知結果を反映して車線を推定する。そして再び終点から順に検索し、車線変更の検知結果を反映することで全区間の車線推定を行う。

3.4 検知処理

実際に収集した走行ログデータから、右左折および車線変更の検知方法の検討を行った。具体的には、走行ログデータの調査のための予備実験について 3.4.1 項で説明し、その結果を受けて 3.4.2 項、3.4.3 項で右左折、車線変更の検知方法について検討を行った。

3.4.1 予備実験

右左折、車線変更検知のためのログデータの波形特徴について調査し、検知手法について検討した。

ログデータ収集にあたって、スマートフォンである Android 端末でロギングアプリケーションを実装した。スマートフォンを図 7 のように画面が上を向くように車内前方のダッシュボードに設置し、走行した際の 3 軸加速度、3 軸方位、GPS の補測値（緯度、経度、高度、速度、方位）を 0.1 秒間隔で記録した。

右左折および車線変更時の各センサデータの分析を行う。まず、右左折時におけ

る加速度センサ、方位センサの値の変化を図 8、図 9 に示す。



図 7 スマートフォンの設置イメージ

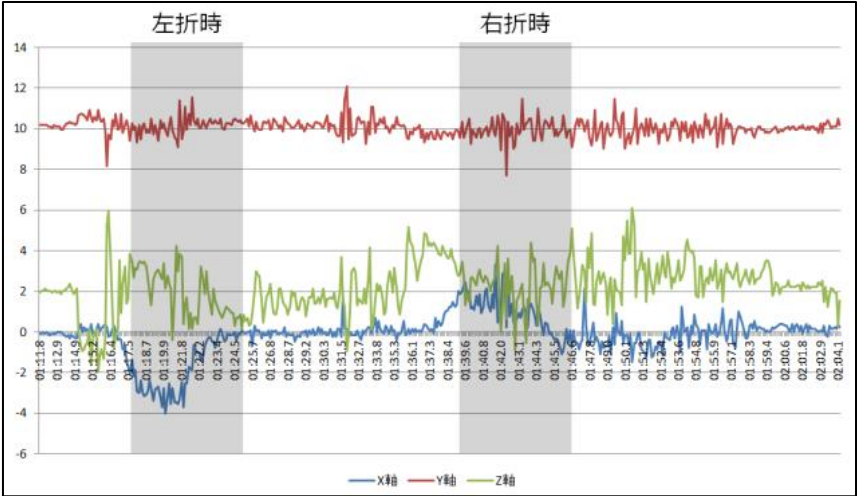


図 8 右左折時の加速度の変化

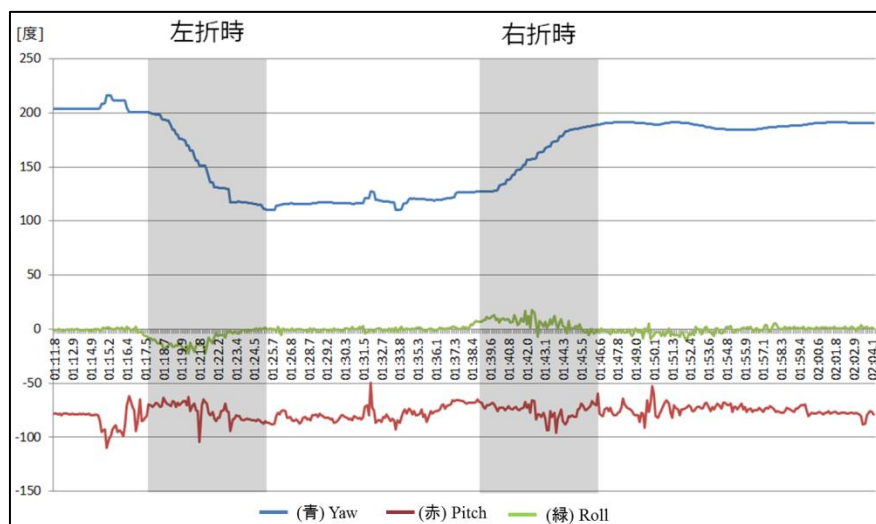


図9 右左折時の方位の変化

最初に、加速度の変化を調査した。図8は走行時の3軸加速度の変化を示しており、影になっている部分が右左折を行った部分である。右左折を行う場合の車両の動きとして、一旦速度を減速させるか信号などで停止した状態から右左折の動きをするのが一般的である。そのため、右左折以前は車両の加速度に変化があると考えられ、図8の右左折の部分でも加速度の値が他の部分と比べて変化しているのがわかる。しかし、加減速を行う場合は右左折時だけでなく、信号での停止時や前方に障害物があり、停止する場合なども当てはまる。そのため、加速度の変化のみでは右左折を特定するのが困難であると考えられる。

そこで、次に右左折時の方位の値の変化を調査した。図9は走行時の3軸方位(yaw, pitch, roll)の変化であり、影の部分が右左折を行った部分である。この図から、右左折時にYaw方向の方位(青線)の値が大きく変化していることがわかる。これは、端末の画面が真上に向くように設置していたため、Yaw方向の値が車体の向きの変化を検知するので、右左折の際にYaw方向が大きく変化していると考えられる。更に、方位の値が大きく変化する場合は、Uターンや右左折時、道路が大きく湾曲している場合などのごく少数のパターンに分けられる。そこで、方位の大きな変化を検知した場合は道路ネットワークデータにその地点の位置情報を問い合わせ、その地点が交差点である場合に右左折を検知すれば検知精度が向上すると考えられる。

以上の結果から、右左折の検知には方位センサのYaw方向の値の変化を利用し、更にその変化を検知した地点の位置情報を道路ネットワークデータの問い合わせ、その地点が交差点であった場合に右左折を検知するという手順で検知する。

車線変更の検知は道路ネットワークデータに登録されている方位の値とログデータの方位の値を比較し、その差異の大きさによって検知する。車線変更を行いながら走行した区間の走行ログデータのうち、車体の向きであるYaw方向の値(青線)と道路ネットワークデータの方位の値(赤線)を比較するグラフを以下に示す(図10)。

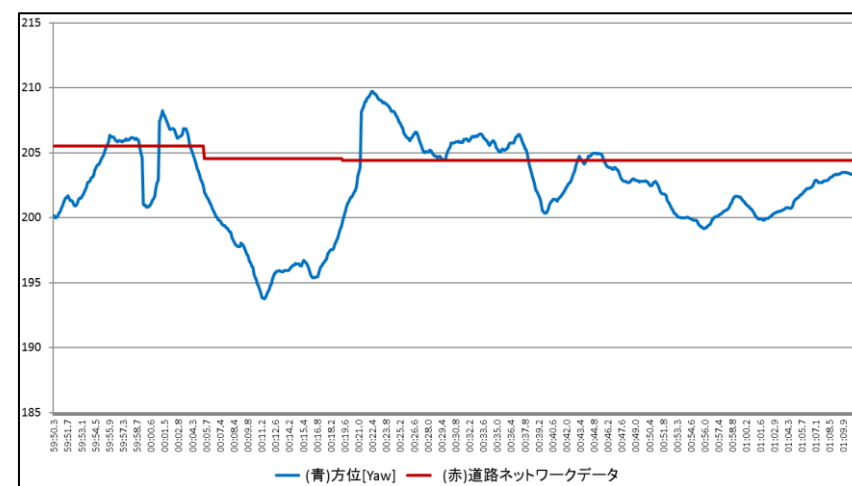


図10 車線変更時の方位の値の比較

道路ネットワークデータには、各セグメントの道路の向きが格納されているため、セグメント内であれば一定の値を示す。そして、各セグメントは実際の道路の交差点と形状が直線である点を結んで格納しているため、道路と平行に走行した際の走行ログデータと道路ネットワークデータに格納している方位の値は大きな差がないと考えられる。図10のうち、影になっている部分が車線変更を行った部分を示している。1回目の車線変更と二回目の車線変更の間で走行ログデータと道路ネットワークデータの方位の値にずれが生じているが、それ以外の部分では大きなずれはないように見える。そこで、走行ログデータと道路ネットワークデータの方位の値の差異をとり、その差異の波形を調査する。差異の値の変化のグラフを図11に示す。

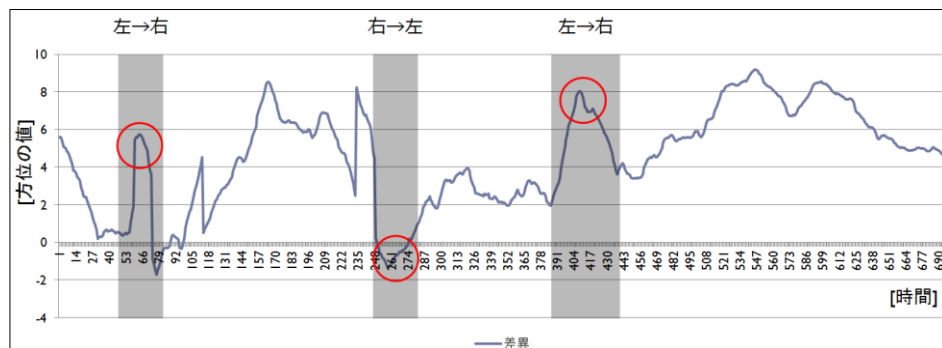


図 11 方位の値の差異

差異は道路ネットワークデータの方位値から走行ログデータの方位値を引くことで算出している。図 11 でも同様に、影になっている部分が車線変更を行った部分である。赤丸で示されている通り、左車線から右車線への車線変更時には山の波形が、右車線から左車線への車線変更時は谷の波形が現れていることがわかる。これは、道路と平行に走行していた車両の車体が車線変更の際に一度傾き、車線変更が完了するときに再び道路と平行になるという車線変更特有の動きのためである。この山、谷の波形を検知することで車線変更地点が検知可能である。

以上の予備実験から得られた結果から右左折、車線変更検知の具体的な処理を実装する。具体的な処理については後述する。

3.4.2 右左折検知処理

走行ログデータの調査結果を受け、まず右左折検知の処理について検討を行った。右左折検知について、検知方法を図 12 に示す。方位センサから取得する値は、端末の向きが右回りに動けば値が増加し、左回りに動けば値が減少する。したがって、方位センサの値がある一定の値以上増加していれば右折、減少していれば左折を検知する。

具体的には、時間窓 $T_1(sec)$ を設け、走行ログデータを時系列順にスキャンしてゆく。スキャン時に T_1 を 2 等分し、各時間窓での平均値を求める。分割窓における平均値を時系列順にそれぞれ平均値 $A_1(deg)$ 、平均値 $A_2(deg)$ とし、右左折の閾値を $T_t(deg)$ としたとき、右左折検知の条件式は以下に示すとおりである(1)。

$$\begin{aligned} \text{右折: } A_2 - A_1 &> T_t \\ \text{左折: } A_1 - A_2 &> T_t \end{aligned} \quad \cdots (1)$$

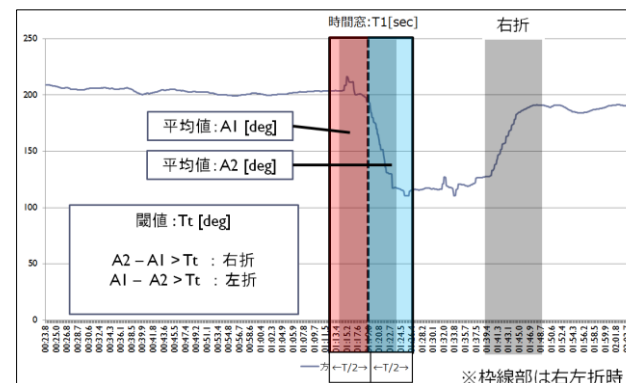


図 12 右左折の検知方法

3.4.3 車線変更検知処理

次に、車線変更の検知手法について説明する。車線変更は 4.2 節でも述べたとおり、走行ログデータと道路ネットワークデータの方位値の差異に山、谷の波形が現れている場合に検知する。検知方法は右左折の検知方法と同様にして、車線変更検知の時間窓 $T_2(sec)$ を設けて走行ログデータを時系列順にスキャンしてゆく。車線変更検知方法の例として、山の検知方法を図 13 に示す。

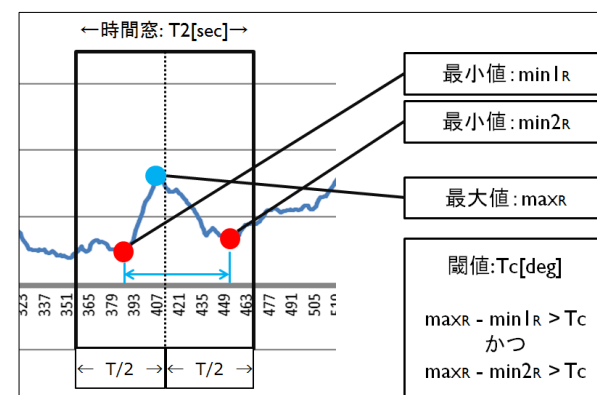


図 13 山の検知方法

山の検知方法として時間窓 T_2 を 2 等分し各時間窓の最小値 ($min1_R, min2_R$) をとる．そして、とった最小値の間の最大値 (max_R) をとる．車線変更の閾値を T_c (deg)としたとき、最大値 (max_R) から各最小値 ($min1_R, min2_R$) を引いた値がどちらも閾値 T_c よりも大きいとき、山を検知する．谷の検知方法も同様に、各分割窓の最大値 ($max1_L, max2_L$) とその間の最小値 (min_L) をとる．各最大値 ($max1_L, max2_L$) から最小値 (min_L) を引いた値がどちらも閾値 T_c よりも大きいとき、谷を検知する．車線変更検知の条件式は以下に示すとおりである．

$$\begin{aligned} \text{右車線への車線変更 (山の検知): } & \max_R - \min1_R > T_c \text{ かつ } \max_R - \min2_R > T_c \\ \text{左車線への車線変更 (谷の検知): } & \max1_L - \min_L > T_c \text{ かつ } \max2_L - \min_L > T_c \quad \cdots (2) \end{aligned}$$

各検知方法のパラメータ設定については 4.3 節で述べる．

3.5 推定処理手順

車線推定処理の全体の流れを図 5 の処理手順に沿って説明する．

提案手法では、道路幅 5m ごとに車線推定を行うため、まず道路ネットワークデータのセグメントテーブルを 5m ごとに分割した分割セグメントテーブルを生成する (①)．次に、走行ログデータを取り込み、分割セグメントテーブルにタグ付けをし、更新する (②)．さらに、走行ログデータの位置情報を道路セグメントテーブルに問い合わせ、道路データの方位値を取得することで、3.4 節で説明した手法を用いて右左折および車線変更を検知する．そして、検知結果を再び分割セグメントテーブルにタグ付けし、更新する．最後に、②の検知結果に基づき、終点からスキャンしてゆき (推定処理の流れは 3.3.3 項参照)、全区間の車線を推定する (③)．以上の結果を経て、分割セグメントテーブルに走行ログデータの車線推定結果が格納される．

4. 実験および評価

本章では、提案システムの有効性の検証のための実験と評価について述べる．実験は、3 章で提案した右左折、車線変更検知のパラメータ設定のための実験と走行した全区間の推定精度の評価実験を行う．

4.1 実装

右左折、車線変更検知および走行区間の車線推定処理は Java とデータベース接続の API である JDBC (Java Database Connection) を用いて実装した．検知結果および推定結果は可視化や評価に利用しやすいように汎用性の高いマークアップ言語であ

る XML (Extensible Markup Language) を用いて出力するようにしている．扱う道路データに関して、位置情報および道路セグメントの長さ、方位はデータベースシステムのオープンソフトウェアである PostgreSQL の地理空間情報を扱うための拡張機能である PostGIS を利用して定義している．セグメントの位置情報は線データ (MULTIPOINT)、中点の位置情報は点データ (POINT) であり、各地点の緯度、経度が格納されている．また、セグメント長は PostGIS の Distance 関数を、方位は Azimuth 関数を利用し、算出した数値をそれぞれメートル単位、度単位で格納している．また、GPS の精度調査および検知結果、推定結果の可視化ためのアプリケーションを JavaScript で実装した．

4.2 実験環境

パラメータ設定実験では、片側 2 車線道路と片側 1 車線道路の混在する約 4km 区間を走行した際のログデータを用いる．この実験では、右折を 2 回、左折を 2 回、車線変更を 2 回行っている．用いる一つのログデータに対し、右左折、車線変更を行った地点での検知処理の各パラメータについて検討する．車線推定精度の評価実験では、右左折および車線変更を数回行いながら走行した 10 個のログデータを用いて、推定精度評価を行う．

4.3 パラメータ設定実験

右左折、車線変更における実験のパラメータを表 3 に示す．なお、検知処理のパラメータである閾値と時間窓はそれぞれ 3.5 節で述べた T_t, T_c である．

表 3 実験パラメータ

	閾値(deg)	時間窓(sec)
右左折	50,55,60,65,70,75,80,85,90,95,100,105,110	1,2,3,4,5,6,7,8,9,10
車線変更	1,2,3,4,5,6,7,8,9,10	1,2,3,4,5,6,7,8,9,10

右左折の閾値は、交差点の形状を考慮して 90 度(deg)近傍を 5 度間隔で 13 個定めており、車線変更の閾値は車線変更の際の車体の傾きの大きさから 1 度(deg)から 10 度(deg)までの値を 1 度間隔で定めた．時間窓は右左折、車線変更の継続時間に影響するため、どちらも 1 秒(sec)から 10 秒(sec)までの間で、1 秒間隔で定めた．また、検知結果の評価指標である検知率を式(3)のように定義する．なお、検知率を $Rdetect$ (%), 正解数を $Ncor$, 全検知数を $Nall$ とする．

$$Rdetect = Ncor / Nall \quad \cdots (3)$$

まず、右左折検知のパラメータについて検討する。右左折のパラメータ設定実験結果を表4に示す。

表4 右左折検知の正答率

時間窓(sec) 閾値(deg)	1	2	3	4	5	6	7	8	9	10
50	0%	20%	60%	80%	100%	100%	60%	60%	60%	60%
55	0%	20%	60%	60%	100%	100%	80%	60%	60%	60%
60	0%	20%	40%	40%	40%	100%	100%	60%	60%	60%
65	0%	20%	20%	40%	40%	60%	60%	60%	60%	60%
70	0%	20%	20%	40%	40%	40%	60%	60%	60%	60%
75	0%	20%	20%	40%	40%	40%	60%	60%	60%	60%
80	0%	20%	20%	40%	40%	40%	40%	60%	60%	40%
85	0%	2%	20%	20%	20%	40%	40%	60%	60%	40%
90	0%	0%	20%	20%	20%	40%	40%	40%	60%	60%
95	0%	0%	20%	20%	20%	20%	40%	40%	60%	40%
100	0%	0%	0%	0%	0%	0%	0%	40%	40%	40%
105	0%	0%	0%	0%	0%	0%	0%	0%	40%	40%
110	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%

表4は、右左折検知の閾値と時間窓別の正答率を示しており、検知率が高い組み合わせのセルを強調して示している。表で示した結果から、閾値が50, 55, 60度(deg)と閾値を小さい値に設定した時に正答率が高くなっていることがわかる。また、時間窓については5, 6, 7秒(sec)付近で検知率が高くなっている。そのため、走行した全区間の車線推定の精度評価実験には閾値が50, 55, 60度(deg)の各値を組み合わせた計9個のパラメータを使用する。

次に、車線推定検知のパラメータについて検討する。車線変更のパラメータ設定実験結果を表5に示す。

表5は、表4と同様に車線変更検知の閾値と時間窓別の正答率を示しており、検知率が高い組み合わせのセルを強調して示している。右左折検知に比べて車線変更検知の検知率は全体的に低い結果となった。その結果の中でも比較的良好な検知率を示しているパラメータは閾値が5, 6, 7度(deg)、時間窓が2, 3, 4秒(sec)の時である。そのため、車線推定の精度評価実験には右左折検知と同様に閾値5, 6, 7度(deg)、時間窓2, 3, 4秒(sec)を組み合わせた計9個のパラメータを使用する。

表5 車線変更検知の正答率

時間窓(sec) 閾値(deg)	1	2	3	4	5	6	7	8	9	10
1	33.3%	13.6%	10.7%	7.0%	5.8%	5.3%	4.5%	5.1%	6.1%	5.3%
2	66.7%	25.0%	17.6%	12.5%	10.0%	8.1%	7.3%	7.1%	7.0%	6.7%
3	33.3%	25.0%	25.0%	15.0%	12.0%	10.7%	9.4%	9.1%	8.8%	8.3%
4	33.3%	42.9%	25.0%	17.6%	15.8%	12.0%	10.7%	1.1%	9.7%	9.4%
5	33.3%	75.0%	50.0%	33.3%	33.3%	12.0%	18.8%	16.7%	15.8%	14.3%
6	33.3%	33.3%	33.3%	66.7%	50.0%	25.0%	40.0%	28.6%	28.6%	28.6%
7	33.3%	33.3%	33.3%	66.7%	33.3%	50.0%	33.3%	33.3%	66.7%	66.7%
8	0.0%	33.3%	33.3%	66.7%	33.3%	33.3%	33.3%	33.3%	33.3%	33.3%
9	0.0%	33.3%	33.3%	66.7%	33.3%	33.3%	33.3%	33.3%	33.3%	33.3%
10	0.0%	33.3%	33.3%	66.7%	33.3%	33.3%	33.3%	33.3%	33.3%	33.3%

4.4 車線推定精度評価実験

4.3のパラメータ設定実験により決定した右左折、車線変更検知のパラメータを用いて全区間の車線推定を行い、その推定精度を評価する。実験には、パラメータ設定実験と同様の条件で収集した走行ログデータを10個用いて、その車線推定精度評価を行う。精度評価において、分割セグメントテーブルに格納される車線推定結果を評価するため、評価の粒度は道路幅5m単位での評価となる。また、右左折および車線変更の検知のタイミングはその動作の開始から終了までの区間の midpoint とする。精度評価実験の評価指標として、式(4)のように正答率を定義する。なお、正答率を $R_{correct}$ 、正解数を N_{cor} 、道路データの分割数を N_{div} とする。

$$R_{correct} (\%) = N_{cor} / N_{div} \quad \cdots (4)$$

実験の結果として、右左折の検知は4.3節のパラメータ設定実験で決定したどの閾値、時間窓においても100%の検知率が得られた。そのため、評価は右左折と車線変更の全パラメータを網羅した結果ではなく、右左折の検知率が100%である場合の車線変更の各パラメータの組み合わせの正答率について評価する。表6に使用したログデータの詳細を示し、図14に精度評価実験の結果を示す。

表 6 ログデータの詳細

ログデータ	1	2	3	4	5	6	7	8	9	10
走行距離	1010m	1910m	1910m	5955m	1010m	2180m	1010m	1010m	1010m	1010m
右左折回数	3回	2回	2回	3回	3回	2回	3回	3回	3回	3回
車線変更回数	2回	5回	2回	4回	2回	3回	2回	2回	2回	3回

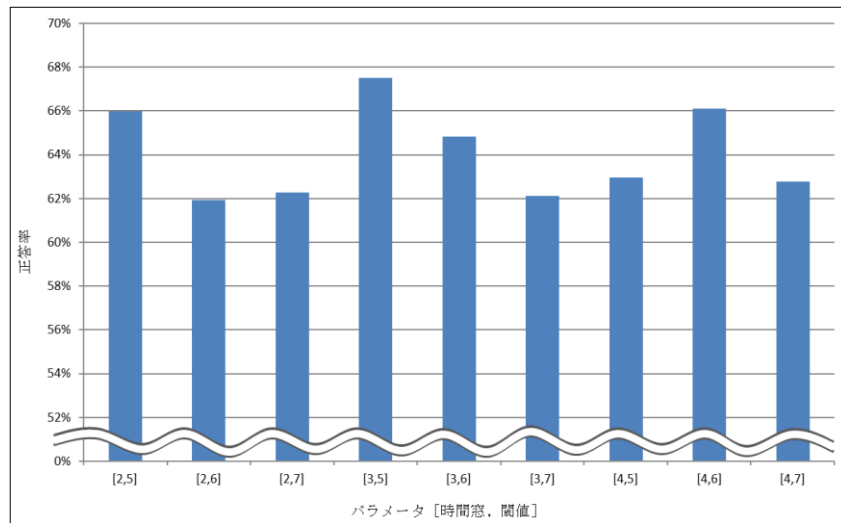


図 14 各パラメータにおける正答率の平均

図 14 は 10 個のログデータの車線推定結果の各パラメータにおける正答率の平均値を示しており、縦軸は推定結果の正答率、横軸はパラメータを示している。結果の中で一番高精度な推定結果が得られたパラメータは閾値が 5 度(deg)、時間窓が 3 秒(sec)の時であり、正答率は 67.5%であった。パラメータと正答率の関係性として、閾値の値が大きく、時間窓が小さいほど正答率が高くなる傾向が得られた。また、ログデータによって車線変更の誤検知数と正答率に大きな違いが見られた。各ログデータにおける車線変更の誤検知数について表 7 に示す。

表 7 車線変更の誤検知数

ログデータ	1	2	3	4	5	6	7	8	9	10
[時間窓, 閾値]	[2,5]	▲0 △15	▲3 △1	▲0 △5	▲0 △334	▲1 △36	▲0 △143	▲0 △29	▲0 △71	▲0 △38 △12
	[2,6]	▲0 △17	▲3 △1	▲0 △5	▲0 △378	▲1 △46	▲0 △161	▲0 △40	▲0 △73	▲0 △45 △13
	[2,7]	▲0 △17	▲3 △1	▲0 △20	▲0 △407	▲0 △52	▲0 △174	▲0 △44	▲0 △73	▲0 △50 △16
	[3,5]	▲0 △13	▲4 △1	▲1 △8	▲0 △265	▲0 △20	▲0 △101	▲0 △25	▲0 △64	▲0 △26 △6
	[3,6]	▲0 △16	▲4 △1	▲1 △9	▲0 △309	▲0 △30	▲0 △120	▲0 △31	▲0 △68	▲0 △35 △6
	[3,7]	▲0 △17	▲4 △1	▲1 △12	▲0 △347	▲0 △40	▲0 △137	▲0 △34	▲0 △69	▲0 △40 △8
	[4,5]	▲0 △7	▲5 △1	▲1 △8	▲0 △220	▲1 △8	▲0 △84	▲0 △15	▲0 △57	▲0 △20 △4
	[4,6]	▲0 △9	▲4 △1	▲1 △8	▲0 △265	▲1 △10	▲0 △95	▲0 △20	▲0 △63	▲0 △29 △7
	[4,7]	▲0 △11	▲4 △1	▲1 △10	▲0 △302	▲1 △14	▲0 △105	▲0 △24	▲0 △65	▲0 △33 △8

▲：false negative による誤検知数

△：false positive による誤検知数

表 7 中の▲は実際に車線変更を行った地点を検知できない false negative の誤検知数を示しており、△は実際には車線変更を行っていない地点を誤検知してしまう false positive の誤検知数を示している。全体的に false positive による誤検知数が多く、それは推定するログデータに依存している。しかし、誤検知数と正答率に比例の関係は見られなかった。また、車線変更誤検知の要因分析のために、実際に車線変更を行った地点と誤検知の地点を図 15 に示す。

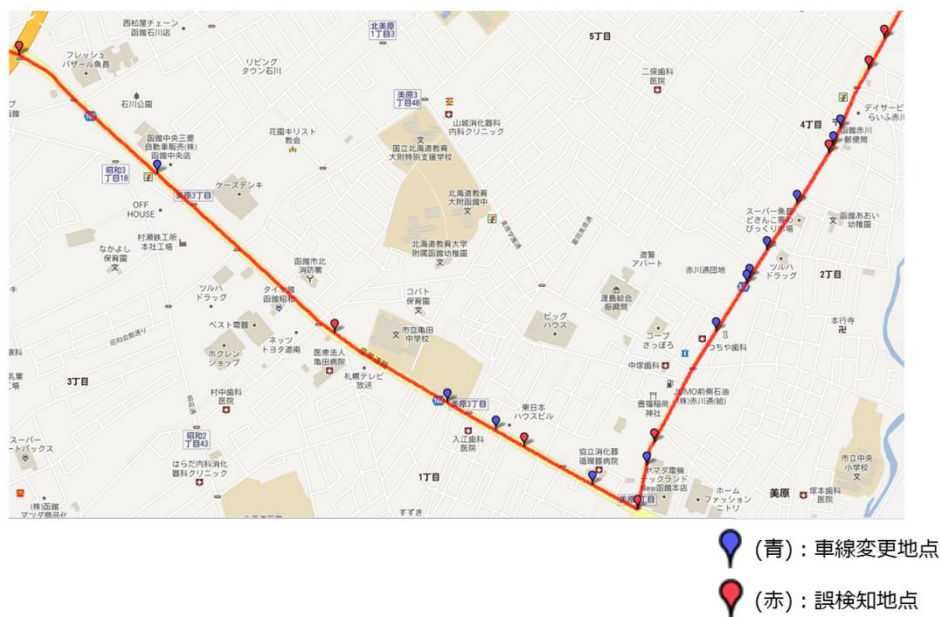


図 15 車線変更地点と誤検知地点

図 15 では、走行した 2 車線区間の経路を赤線で示しており、実際に車線変更を行った地点を青いピン、10 個のログデータのうち、3 回以上誤推定をした地点を赤いピンで示している。誤検知は、交差点や道路が湾曲している地点など車線変更を行わずに車両の向きが変化する点で多く検知されている。また、ある地点で誤検知をするとその近辺で連続的に誤検知をするという特徴が得られた。

4.5 考察

本節ではパラメータ設定実験および車線推定精度評価実験における実験結果について考察を行う。

まず、右左折の検知はパラメータ設定実験で設定したパラメータが妥当であることが精度評価実験の結果から確認できた。次に、車線変更の検知は右左折の検知に比べて良好な検知率が得られたパラメータはなく、精度評価実験でも false positive による誤検知が多く検出された。その原因として、道路ネットワークデータの構造上、セグメントの境界で方位値ログデータの差異の波形に大きな影響が出ていることが考えられる。道路ネットワークデータは湾曲地点や交差点で区切ることで、セグメントを構

成しているため、その境界の方位値は急激に変化しており、方位ログデータとの差異に急激な変化が現れてしまう。また、車線変更は運転や道路形状に影響を受けていることも検知率が低い原因であると考えられる。急な車線変更を行う場合と滑らかに車線変更を行う場合では、方位の波形特徴に大きな違いが出ると考えられる。また、湾曲している道路に沿って車線変更を行った場合は、方位ログデータの波形に山や谷の波形が現れないと考えられる。精度向上のためには車線変更の特徴や道路の形状によって車線変更検知のパラメータを動的に最適値に変更する必要があると考えられるが、全ての道路形状や車線変更の特徴に適応する最適値を定めるのは難しい。そのため、GPS の補測値である位置情報の利用や道路セグメントの境界の方位値を補正するなど、精度向上のための手法の検討が必要である。

最後に、全区間の車線推定の実験結果では全体の正答率の平均は 60% から 70% の間であり、高精度な推定結果は得られなかった。その原因として、車線推定処理が車線変更の検知結果に依存してしまっていることが問題であると考えられる。3.3.3 項で述べた車線推定手順は右左折検知結果を反映後に車線変更検知結果を反映する手順であるため、全体の車線推定結果は車線変更の検知結果の依存度が高い。さらに、車線変更の検知率が低いにもかかわらず、その検知結果をもとに車線推定を右車線か左車線のどちらかに決定してしまっていることが、精度が低い原因であると考えられる。

そのため、車線変更の推定精度向上に向けて車線変更の検知率の向上および右左折の検知結果を優先した車線推定アルゴリズムを検討していく必要がある。また、検知する車両の動作（直進、右左折、車線変更）の信頼性は検知率や動作を行う環境に依存するため、検知した状態によって車線推定を確率的に行う手法も有効であると考えられる。具体的には、文献[17]で採用していた隠れマルコフモデルも全区間の推定精度向上に有効であると考えられるため、今後の検討課題である。

5. まとめ

本稿では、スマートフォンに組み込まれているセンサ機器を利用することで従来手法よりも計算コスト、導入コストを削減したロバスト性の高い車線推定手法を提案した。また、提案手法をスマートフォンに実装し、右左折、車線変更の検知率および車線推定結果の精度について実験、考察を行った。今後は車線推定精度向上に向けた車線変更検知の補正手法を検討するとともに右左折の検知結果の優先度を高めた車線推定アルゴリズムを検討していく。また、片側 3 車線道路や右左折専用レーンなど、特殊な環境下における車線推定システムの検討も今後行いたいと考えている。

参考文献

- [1] 和田光示, プローブ情報システム(IPCar)プロジェクト, 情報処理 Vol.43, No.4, pp.363-368, 2002.
- [2] 村田利幸, カーナビゲーションにおけるプローブ情報の活用, JACIC 情報 Vol.25, No.2, pp.57-60, 2010.
- [3] 植原啓介, プローブ情報システム : 車載センサを活用した環境情報の取得, 情報処理 Vol.51, No.9, pp.1144-1149, 2010.
- [4] 渡辺恭人, 交通の IT 化とプローブ情報の利活用, CUC view & vision 27, pp.43-47, 2009.
- [5] 森川高行, プローブ情報を利用した道路交通サービスの可能性, システム制御情報学会誌 Vol.54, No.9, pp.366-370, 2010.
- [6] VICS Web Site, 財団法人道路交通情報通信システムセンター, <http://www.vics.or.jp/index1.html>, 2011.
- [7] 柴崎裕昭, カーナビゲーションにおける ITS 対応の動向, 自動車技術 Vol.65, No.2, pp.80-83, 2011.
- [8] 前中一介, MEMS センサ MEMS 技術の動向と応用, 科学と工業 Vol.85, No.2, pp.57-67, 2011.
- [9] 檜橋祥一, 岡崎浩司, 福田敦史, 携帯電話と MEMS 技術, マテリアルステージ Vol.8, No.10, pp.28-33, 2009.
- [10] 安本慶一, アーメドアサド, Participatory Sensing の研究動向, 電子情報通信学会研究報告.IN, 情報ネットワーク, Vol.111, No.245, 2011.
- [11] 戸辺義人, 都市レベルセンシングネットワーク : OSOITE プロジェクトの経験と今後, 電子情報通信学会研究報告.USN, ユビキタスセンサネットワーク, Vol.111, No.54, 2011.
- [12] ディファレンシャル GPS とは, 海上保安庁 DGPS センター, <http://www.kaiho.mlit.go.jp/syokukai/soshiki/toudai/dgps/index.htm>, 2011.
- [13] トランジスタ技術編集部, GPS のしくみと応用技術ー測位原理, 受信データの詳細から応用製作まで, CQ 出版, 東京, 2009.
- [14] A-GPS とは, an Impress Group company, http://k-tai.impress.co.jp/docs/column/keyword/20100223_350721.htm, 2011.
- [15] 西村明彦, 金帝演, 長谷川孝明, 複車線における M-CubITS での車両位置特定について, 映像情報メディア学会技術報告, Vol.33, No.6, pp.115-119, 2009.
- [16] 西脇由博, 宮島千代美, 北岡教英, 武田一哉, 確率的手法を用いた車線変更軌跡のモデル化, 情報処理学会論文誌 Vol.51, No.1, pp.131-140, 2010.
- [17] 斉藤裕樹, 菅生啓示, 間博人, テープウィロージャナポン ニワット, 戸辺義人, sBike : 参加型センシングを志向したモバイルセンシングによる自転車走行状態収集・共有機構, 情報処理学会論文誌, Vol.53, No.2, pp.770-782, 2012.
- [18] CARS JAPAN Data base of car performance & specs, Cars Japan, <http://cars.jpn.org/index.html>, 2012.