

網情報を学習に用いた 鉄道利用中の高精度な経路推定手法の提案

神保 拓也^{1,a)} 藤波 香織^{2,b)}

概要：日本の公共交通機関は他国と比較して路線数や利用者数が多く、初めて利用するユーザにとっては利用が難しいものとなっている。本稿ではユーザの乗り間違い行動に着目し、駅の位置情報と端末から取得した位置情報によって乗車中のユーザの次の到着駅の予測手法について提案を行う。正確な到着駅判定を行うため、路線情報を学習に用いた予測モデルの提案を行った。具体的に、移動経路上の各点と各駅との関係をあらかじめ学習することによって同様のデータが与えられた際の判定精度の向上を図る。精度評価において、370 駅分の区間データを用いて先行研究の手法との比較を行った結果、先行研究 (73.9%) よりも約 22 ポイント精度が向上した。また、3 種類の特徴量の組み合わせのうち、リンク特徴量のみを学習した最近傍法による判定結果 (95.6%) が最大となった。

1. はじめに

近年、スマートフォンをはじめとするモバイル端末の普及により、端末上でのナビゲーションシステムが多く開発されるようになった。それらのシステムの多くは、端末の GPS と地図情報を組み合わせたルート案内システムである。そのようなルート案内システムは一般的に、ユーザが自ら確認を行う必要があり、確認に基づいて最終的な判断はユーザ自身によって下される。図 1 と図 2 はそれぞれ Yahoo!路線情報 [1] のルート検索結果と GoogleMap[2] による地図表示画面である。Yahoo!乗換案内では、時刻表にもとづく現在位置がリアルタイムに表示される。また、目的の駅の到着前にユーザに対してアラートする機能が存在する。これに対し、GoogleMap では GPS から取得した位置情報を地図上にプロットすることにより現在地点の情報表示を行っている。これらのシステムはユーザが表示情報から現在の状況を判断する必要があり、不慣れな利用者にとっては利用が難しいものと考えられる。Yahoo!乗換案内のサービスではアラート機能や、時刻表に基づく現在位置が表示されるが、これはあくまで正しい列車に乗車して

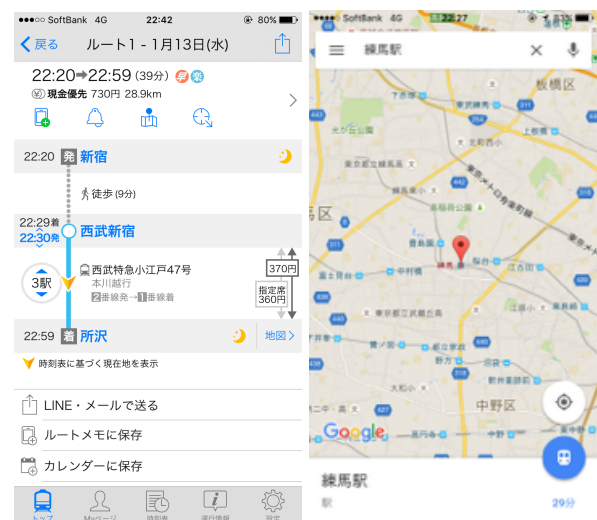


図 1 Yahoo!路線情報

図 2 Google Map

いた場合について有効であり、その前に乗り間違いをしていた場合には全く役に立たないものとなる。ユーザに対してより密接なナビゲーションを実現するためには、システムが自動的に在線状況を推定し、必要に応じて情報提示を行うことが望ましいといえる。筆者らはこれまでに、乗り間違いに着目した自動的な誤乗車検出システムを提案した [3]。評価の結果、誤乗車検出部分の検出精度が十分でないため、誤判定が増加してしまうといった問題点が見られた。本稿では経路推定部分について、地理的な鉄道網情報を学習することによる到着駅判定手法を提案し、検出精度の高精度化を図る。

¹ 東京農工大学大学院 工学府 情報工学専攻
Department of Computer and Information Sciences, Tokyo University of Agriculture and Technology

² 東京農工大学大学院 工学府 先端情報科学部門
Department of Computer and Information Sciences, Tokyo University of Agriculture and Technology

a) tarohatu@gmail.com

b) fujinami@cc.tuat.ac.jp

2. 関連研究

2.1 リアルタイム GIS による詳細情報提供システム

近年、バスや鉄道車両のリアルタイム位置情報の提供 [4][5] が行われるようになり、それらを利用した研究 [6][7] やアプリケーション開発が進められている。また、2016 年 1 月現在山手線全線で施行されている「山手線トレインネット」[8] では車両ごとに個別 WiFi システムを搭載し、列車の混雑状況や室温など様々な情報をユーザに提供することが可能となっている。これらの取り組みは、これまでと比較してより詳細な車両に関する情報を提供することができる反面、敷設・整備コストの増大といった懸念点も存在する。またリアルタイム GIS の提供は交通機関の企業単位で行われることが多く、多くの路線が乗り入れるターミナル駅などでは複数の路線に関する情報提示が困難であると推測される。以上を踏まえ本稿では、モバイル端末から取得する位置情報以外のリアルタイムなデータを使用せずに、次の到着駅予測の高精度化を目指す。

2.2 これまでの取り組み

筆者らはこれまでに、慣性センサと GPS を組み合わせた自動的な誤乗車検出システムの開発を行った [3]。先行研究 [3] において、GPS から取得した位置情報と各駅の位置情報から相対ベクトルのコサイン類似度計算を行い、その大きさによって次の到着駅の予測を行っている。提案システムによる実験において、カーブや駅の密集したターミナル駅のような場所から出発する場合に判定精度が極端に低下するといった問題点が見られた。図 3 は先行研究における JR 山手線池袋一大塚駅間を移動した際の判定結果の遷移を表している。判定はベクトルの計算に基づいて行われるため、出発開始時の曲線の延長上に存在する北池袋駅への誤判定が増加している。類似した特徴を持つ他の駅においても同様の傾向が確認された。これを受け、本稿では誤乗車判定部分の高精度化に主眼を置き、複雑な路線形状に対して高精度に判定を行うための手法について提案を行う。



図 3 先行研究 [3] における池袋駅周辺の路線図

3. 提案手法

3.1 手法概要

本研究では、モバイル端末から取得した位置情報をもとに、次の到着駅を予測することを目的とする。2.1, 2.2 節で述べた関連研究の問題点を改善するため、本稿では以下の要件を満たすように提案手法の設計を行う。

- 駅および路線を構成する地物情報を使用する
- 端末から取得するデータは位置情報のみとする

2.1 節のリアルタイム GIS を用いた研究では、データ提供を行うための環境整備コストがボトルネックになると述べた。筆者らはより簡便な到着駅予測を行うため、無償で提供される静的な GIS データに着目した。静的な GIS データとは駅や路線の位置情報、すなわち緯度と経度を意味しており、これらと任意の方法によってモバイル端末から取得した緯度・経度情報との組み合わせによって到着駅予測を行う。

3.2 網情報の学習

本稿では高精度な到着駅予測のため、予測手法として機械学習を導入し、到着駅の予測モデルの学習を行う。予測モデル学習時の分類対象となるクラスは、ある駅を中心とした時の各隣接駅とする。すなわち本稿で提案する予測手法は、直前の出発駅の情報およびモバイル端末から取得した乗車中の車両の位置情報を入力とし、次の到着駅の予測を行うことを目的とする。予測モデルは駅単位で作成され、ある駅に到着するとその駅に対する予測モデルに切り替えられ、それぞれの隣接駅の中から走行中の車両が次に到着する駅の推定を行う。2.2 節の問題を改善し、進行方向判定における高精度化を実現するため、本稿では新たに網情報の学習という概念を導入する。すなわち、鉄道網上の各点と駅との関係を機械学習によって学習を行うことにより、類似したデータが与えられた際の識別精度の向上を図る。網情報とは、駅（ノード）と駅を両端とする線路（リンク）によって構成される位置情報の集合であり、リンク上の各点を $l \in L$ として表現する。図 4 左は網情報におけるノードとリンクの関係を表している。このうち、直前に出発した駅を $master(m)$ 、 $master$ に隣接する i 番目の駅を $branch(b_i)$ として定義する。リンク上の各点 l に対する相対ベクトル $\vec{l} = \vec{l} - \vec{m}$ と、 $branch$ と $master$ との相対ベクトル $\vec{b}_i = \vec{b}_i - \vec{m}$ の関係を図 4 右に示す。これらの関係から、予測モデルの学習に有効な特徴量の計算を行う。次節では網情報のうち、リンクおよびノードの位置情報を活用した特徴量について定義を行う。

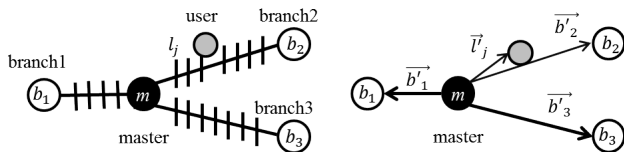


図 4 左：ノードとリンク間の関係図，右：ノードとリンク上の点の相対ベクトルの関係

3.3 学習に使用する特徴量

3.3.1 リンク情報に基づく特徴量

リンクは実際に車両が走行する線路上の位置情報であり，リンク上の各点の緯度・経度と次の到着駅との関係を学習することによって分類を行う．リンクはノードを両端とした有限区間であり，線路上の位置情報を学習することは分類精度の向上に有効であると考えられる．リンク上の各点 l に対し 3.2 節で定義した相対ベクトル $\vec{l}$ をリンク特徴量として定義し，学習データとして使用する．

3.3.2 ノード情報に基づく特徴量

予測先の隣接駅の位置情報を利用したノード由来の特徴量を考えることも可能である．リンク特徴量は直前の出発駅とリンク点 l のみとの関係性を表現するものであった．これに対しノード特徴量は，隣接する各駅 b_i とリンク上の点との関係性を表現する．具体的には，前節で計算した相対ベクトル $\vec{l}, \vec{b}$ を用いて，リンク上の点と，ノード間の関係を表現し，これをノード特徴量として定義する．先行研究において提案を行った，コサイン類似度を使用した手法 [3] の識別規則は次の式で与えられる．

$$next = \arg \max_i \cos \theta_i = \arg \max_i \frac{\vec{l} \cdot \vec{b}_i}{|\vec{l}| |\vec{b}_i|} \quad (1)$$

式 (1) による予測では直線同士の比較を行うため，カーブなどリンクの形状によって誤った予測結果を与えてしまうことが先の研究より明らかとなった．これに対して提案手法では，リンク上を移動する際に取得した位置情報と式 (1) の $\cos \theta$ の関係をあらかじめ学習することで，同様のデータが与えられた際に正解するように予測モデルを作成する． $\cos \theta_i$ に加え，リンク上の各点と隣接駅との距離関係を表現するため，相対ベクトル $\vec{l}, \vec{b}_i$ のノルム比 nr_i を特徴量として定義する．ノルム比の計算式を式 (2) に示す．

$$nr_i = \frac{|\vec{l}|}{|\vec{b}_i|} \quad (2)$$

以上より， $\cos \theta_1, \dots, \cos \theta_k, nr_1, \dots, nr_k$ をノード特徴量として定義し，予測モデルの学習に使用する． k は i 番目の駅に対する総隣接駅数を表すものとする．

4. 評価実験

4.1 評価概要

提案手法の評価を行うため，国土交通省が提供する国土

数値情報 [9] の路線データを使用して分類を行った．路線データは路線ごとに位置情報が格納されており，shp ファイル形式で提供されている．本稿では，路線データを駅の区間ごとに分割し，ある駅を中心とした区間データを，1 つの分類器の学習データとして使用する．評価実験では，JR 東日本，JR 西日本が公開する 2014 年度の乗り入れ客数各 50 位 [10][11] のうち，国土数値情報の路線内番号の並びが適切であった 85 駅分の分類モデルを作成し，精度評価を行った．モデルの隣接駅数，すなわち分類クラス数と分類クラス数ごとに作成した分類モデルの関係を表 2 に示す．各分類モデルの作成にあたり，国土数値情報から取得した区間データを学習データとして，分類モデルの構築を行う．実環境での利用を想定し，モバイル端末から取得したデータを分類することを想定し，もとの区間データに対して平均 0[m]，標準偏差 10.2[m] の正規分布に基づく誤差モデル [12] を加えたものをテストデータとして用いている．評価指標には適合率 (precision) と再現率 (recall) の調和平均である F 値を用いた．F 値の計算方法を式 (3) に示す．

$$Fmeasure = \frac{precision + recall}{2 \times precision \times recall} \quad (3)$$

4.2 評価結果

4.2.1 特徴量ごとの評価結果

3.3 節の計算方法に基づき，各 l に対して特徴量を計算し，予測モデルの学習を行う．その際，学習に使用する特徴量として以下の組み合わせを検証する．

- (1) リンク特徴量のみ (2)
- (2) ノード特徴量のみ (2k)
- (3) リンク特徴量+ノード特徴量 (2+2k)

各項目の括弧内の数字は学習時の特徴量数を表しており， k は前節と同様各モデルの隣接駅数とする．すなわち，隣接駅数が 4 の時のノード特徴量数は $2 \times 4 = 8$ となる．上記の組み合わせごとに分類器の学習を行い，分類結果の比較によって，予測に有効となる特徴量の検討を行う．分類アルゴリズムは最近傍法，ロジスティック回帰，ランダムフォレスト (木の数 100) をの 3 種類を用いて，学習を行った．その結果，表 1 より，リンク特徴量のみを学習させた最近傍法による識別が最も高精度となった．

表 1 使用特徴量ごとの評価結果 (F 値)

	リンク	ノード	ノード+リンク
最近傍法	0.956	0.948	0.949
ランダムフォレスト	0.935	0.928	0.935
ロジスティック回帰	0.315	0.768	0.768

表 2 隣接駅数ごとに作成したモデル数

隣接駅数 (駅)	2	3	4	5	6	7	8	9
作成したモデル (個)	20	10	19	15	9	4	5	3
学習データ数の平均 (個)	49.2	92.0	109.0	116.8	114.4	112.7	181.0	137.6
学習データ数の標準偏差 (個)	24.4	34.0	37.7	71.7	52.0	48.0	40.6	16.9
テストデータ数の平均 (個)	422.3	643.9	880.4	1144.4	1326.1	1593.0	1926.0	1979.3
テストデータ数の標準偏差 (個)	17.0	22.2	97.8	96.0	215.0	167.6	227.4	81.8

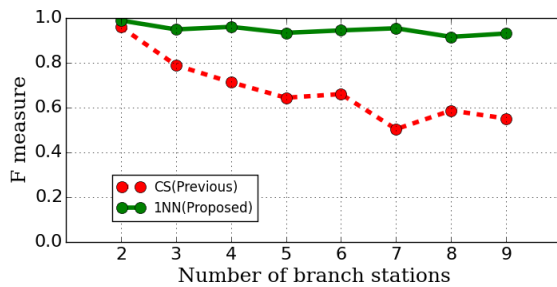


図 5 先行研究 [3] との比較結果

4.2.2 先行研究 [3] との比較

4.2.1 節の結果のうち、最近傍法によるリンク特徴量の学習の結果を、4.1 節のデータを先行研究の手法 [3] (CS) に適用した結果と共に図 5 に示す。前述のとおり、提案手法 (1NN) の F 値平均は 0.956 となり、CS で得られた平均予測精度 (0.739) と比較して 0.22 の精度向上が確認された。各モデルとも隣接駅数の増加とともに予測精度は低下するが、提案手法の F 値平均の最小値が 0.915 (隣接駅数 8) となり、CS と比較して提案手法が駅数の増加に対してより頑健に判定を行うことが可能であるといえる。以上の結果より、網情報、特にリンク情報の学習が予測精度の向上に有効であることが明らかになった。

5. 考察

5.1 特徴量ごとの評価結果に関する考察

4.2.1 節の評価結果において、リンク特徴量のみを学習した最近傍法による分類結果が最高となった。推測される要因として、ノード間のリンクは線分であり、実際の車両もリンク上を走行するため、有限区間で探索をする最近傍法のアルゴリズムが有効であった可能性が推測される。これはノード特徴についても同様であり、車両の線路上での反復移動という特性と、特徴に対して識別空間を計算しない最近傍法の相性によるものと考えられる。一方、ロジスティック回帰は最も低精度であったが、結果より、リンク特徴よりもノード特徴の利用が有効であることがわかった。リンク特徴量において精度が著しく低下した原因として、平面上のリンク特徴の線形分離が困難であるためであると考えられる。リンク上の各点は直前の出発駅を中心と

いて網目状に広がっており、単純な分離平面を引くことができないためであると推測される。

5.2 提案手法の問題点

分類における傾向を調査するため、最近傍法による分類精度の詳細について図 6 に隣接駅数ごとの最近傍法の結果を示す。各隣接駅数における矩形内部の実線部は中央値を表しており、隣接駅数の増加とともに中央値は次第に低下していることが確認できる。しかし、全体を通して分類精度のばらつきは少なく、どの駅に対しても一定の精度での分類が行われている事がわかる。ただし、図 6 の隣接駅数が 3 の場合など、識別精度の下限値が 0.80 を下回っている例も存在することが判明した。

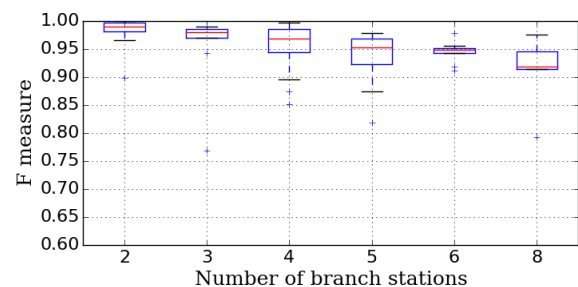


図 6 分類結果の箱ひげ図 (隣接駅数 7 および 9 の結果はデータ数不足のため割愛)

判定精度が低下した駅の性質を調べるため、全判定結果のうち判定精度が最低となった JR 東日本中央線および東京メトロ東西線中野駅に関する位置情報を図 7 に示す。中野駅の隣接駅数は 3 駅と少ないが、中野一東中野、中野一落合区間のうち途中まで線路が重なっている区間が存在する。最近傍法による判定では線路が重なる区間において誤判定が増加し、判定精度が低下したものと推測される。

実際に誤判定が生じた区間を特定するため、提案手法によって誤判定された地点を赤太点で表した結果を図 8~10 に示す。なお各図中の Master および Branch は直前の出発駅および隣接駅を表しており、出発駅から各隣接駅へ進行するものと仮定する。提案手法の判定結果において、図 8~10 より線路の共通する部分において誤判定が増加して



図 7 中野駅の路線図. JR 中央線東中野駅および東京メトロ東西線落合駅のうち、途中までの区間で線路に重なりが存在する。

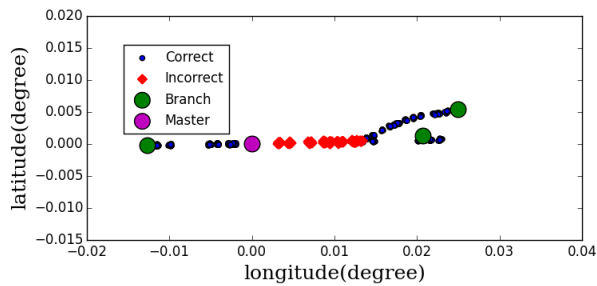


図 8 中野駅を中心とした分類結果

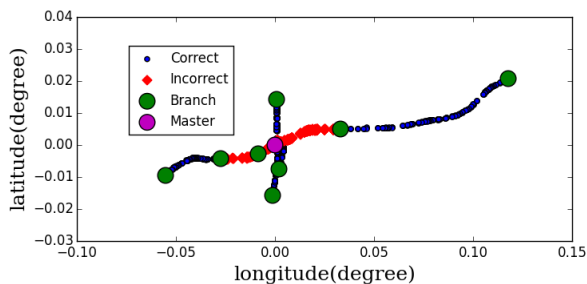


図 9 新鎌ヶ谷駅を中心とした分類結果

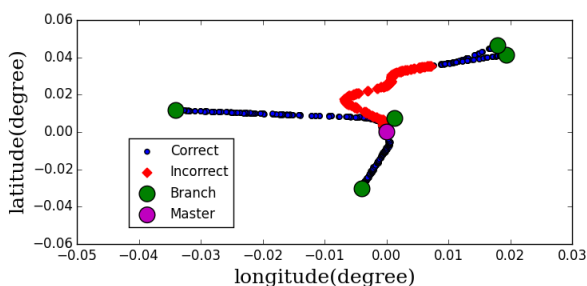


図 10 福島駅を中心とした分類結果

いることが確認できる。それ以外の部分に対する誤判定は減少しており、線路の共通部分での誤判定が全体の判定精度に影響をおよぼす要因となり得るといえる。

6. 今後の取り組み

6.1 問題点の改善

5.2 節では線路の重なりが存在する区間において誤判定が増加するといった問題点が確認された。この問題点を改

善する方法として、路線同士の重なりが存在する駅に対して予測を行う場合においてのみ、軌跡データに対する予測モデルを定義する必要があるといえる。すなわち、複数回の予測結果によって最終的な判定を得ることにより、判定精度の向上を図る。例として、データサイズが4となる軌跡データが与えられた時の中野駅の予測モデルの出力結果が（東中野，落合，東中野，東中野）となり、正解が東中野駅であった場合にこれを学習することで遷移モデルの学習を行い、軌跡データの予測モデルを新たに作成する。また図 11 より、JR 東日本新鎌ヶ谷駅に隣接する新京成線北初富駅と北総線大町駅などのように、ある駅の区間が別の区間に近接し含まれてしまう場合に対しては、時刻表などのより詳細な GIS データを使用することが望まれる。しかし、時刻表は遅延などのダイヤの乱れに対応することが困難であると予想される。

これに対しては、車両の速度を計測することで改善することが可能であると推測される。図 11 より、線路全体が別の路線に近接する区間（以後サブ区間と定義する）を走行する際には、近接する線路間での走行時の加減速のパターンが異なると予想されるため、これを利用することで詳細な到着駅判定が可能になると考えている。

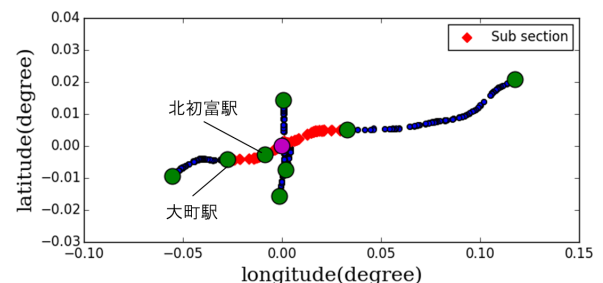


図 11 新鎌ヶ谷駅を中心とした場合のサブ区間の例

6.2 位置情報取得方法の検討

モバイル端末における位置情報の取得方法として GPS による位置情報取得が一般的であり、本稿においても暗黙

的に GPS のデータを仮定して評価を行った。しかし、GPS はその性質上、屋内や地下などの環境下では正確な測位が困難になるといった問題点が存在する。これに対し、それらの環境下での位置情報の取得手段の例として、無線 LAN を用いた研究 [13][14] や、GPS よりも電力効率の高い慣性センサを用いた地下鉄車両の発着判定に関する研究 [15][16] も多く行われており、これらの位置情報取得手法を組み合わせることにより、将来的に地下鉄などの GPS 測位不可能な場所においても本手法が適用可能となると考えられる。

6.3 連続的なトラッキングの実現

以上を踏まえ、今後の取り組みとして連続的な経路推定システムの提案を行う必要がある。本稿で提案した手法では、直前の出発駅が既知であるという前提を与えたが、実際の使用の際には、直前の出発駅を常に更新することが必要である。先行研究 [3] では初期位置となる駅の情報のみを与え、それ以降は相対ベクトルのユークリッド距離の比較によって動的に直前の出発駅の更新を行っていた。具体的には、正解となった駅 b_i とのノルム比が $nr_i \geq 1$ を満たすときにのみ駅の更新を行うことで、各地点において予測を行っていた。今後本研究においてもこれを踏襲し、連続的なトラッキングシステムの構築を目指す。

7. おわりに

本稿では鉄道乗車中の次の到着駅予測を目的とし、予測精度向上のために網情報の学習による分類モデルの構築方法について示した。学習に使用する特徴量として、リンク由来およびノード由来の特徴量を新たに定義し、それらを学習することにより分類精度の向上を図った。評価実験において、370 駅の区間データを用いた精度評価を行い、先行研究 [3] で提案した手法と比較して F 値において 0.22 の精度向上を達成した。今後の取り組みとして、前述の問題点を改善したうえで、次の機能を実装したシステムの開発を行う。

- 位置情報にもとづいた在線位置の判定部分
- 在線位置に応じた誤乗車検出機能

上記の 2 点の実現を本研究における最終的な目的とする。具体的には、モバイル端末から取得した位置情報を利用し、本稿で提案した予測モデルを搭載した在線位置の判定機能、および判定結果を用いた誤乗車検出機能の実装を予定している。またデータセット作成の都合上、本稿では鉄道のみを対象として議論を進めたが、鉄道に並び代表的な公共交通機関であるバスに対する本手法の有効性についても評価を行う予定である。

参考文献

- [1] YAHOO!JAPAN. Yahoo!路線情報: 乗換案内、時刻表、運行情報. [http://transit.yahoo.co.jp/\(2015/11/15アクセス\)](http://transit.yahoo.co.jp/(2015/11/15アクセス)).
- [2] Google. Google マップ. [https://www.google.co.jp/maps/\(2015/11/15アクセス\)](https://www.google.co.jp/maps/(2015/11/15アクセス)).
- [3] Takuya Jimbo and Kaori Fujinami. Detecting mischoice of public transportation route based on smartphone and gis. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2015 ACM International Symposium on Wearable Computers*, pp. 165–168. ACM, 2015.
- [4] Chicago transit authority. [http://www.transitchicago.com/\(2015/1/5アクセス\)](http://www.transitchicago.com/(2015/1/5アクセス)).
- [5] 東京地下鉄株式会社. 東京メトロの「オープンデータ活用」の取り組み. [http://www.soumu.go.jp/main_content/000372119.pdf\(2015/1/5アクセス\)](http://www.soumu.go.jp/main_content/000372119.pdf(2015/1/5アクセス)).
- [6] 伊藤昌毅, 見生元気, 瀬崎薫. 地下鉄乗客のためのスマートフォン内蔵センサを用いた位置特定手法. 電子情報通信学会, ヒューマブローブ研究会 (HPB), 2015.
- [7] Leon Stenneth, Ouri Wolfson, Philip S Yu, and Bo Xu. Transportation mode detection using mobile phones and gis information. In *Proceedings of the 19th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pp. 54–63. ACM, 2011.
- [8] 松本貴之. 鉄道車両内でのパーソナルな情報提供システム. 情報処理学会第 75 回全国大会, Vol. 6, p. 3, 2013.
- [9] 国土交通省. 国土数値情報ダウンロードサービス. [http://nlftp.mlit.go.jp/ksj/\(2015/11/15アクセス\)](http://nlftp.mlit.go.jp/ksj/(2015/11/15アクセス)).
- [10] JR 東日本. 各駅の乗車人員 2014 年度. [http://www.jreast.co.jp/passenger/\(2016/1/3アクセス\)](http://www.jreast.co.jp/passenger/(2016/1/3アクセス)).
- [11] JR 西日本. なんでもランキング. [https://www.westjr.co.jp/railroad/fan/ranking/\(2016/1/3アクセス\)](https://www.westjr.co.jp/railroad/fan/ranking/(2016/1/3アクセス)).
- [12] 坂井丈泰. GPS/GNSS の基礎知識. GPS シンポジウム 2003, 2003.
- [13] 河口信夫. Locky.jp: 無線 LAN を用いた位置推定とその応用. 信学技報, ITS2007-16, July, 2007.
- [14] 安齋恵一, 岡島匠吾, 坪川宏ほか. スマートフォンを用いた屋内位置の推定と歩行ナビゲーションシステム. マルチメディア, 分散協調とモバイルシンポジウム, pp. 921–927, 2011.
- [15] Thomas Stockx, Brent Hecht, and Johannes Schöning. Subways: towards smartphone positioning in underground public transportation systems. In *Proceedings of the 22nd ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pp. 93–102. ACM, 2014.
- [16] Takamasa Higuchi, Hirozumi Yamaguchi, and Teruo Higashino. Tracking motion context of railway passengers by fusion of low-power sensors in mobile devices. In *Proceedings of the 2015 ACM International Symposium on Wearable Computers*, pp. 163–170. ACM, 2015.