

移動支援システムのための移動特性抽出の自動化

鈴木 和希¹ 坂本 大輔² 安積 卓也³ 望月 祐洋⁴ 西尾 信彦²

概要: 近年、スマートフォンの高機能化に伴い、様々なセンサが搭載されるようになってきた。搭載されているセンサを利用することで、端末の位置情報、加速度センサ情報、周辺の Wifi 基地局情報などといったライフログが収集できる。このように収集したライフログを利用することで、ユーザの状況の推定し、ユーザ個人に特化したサービスの提供が行われている。ユーザに個人特化した移動支援に関しても盛んに研究されているが、これらの研究は移動手段の誤認識があったり、対応していない移動手段があったりと問題がある。そこで本研究では、移動支援システムのために移動特性を抽出する。この移動特性を抽出し、同一の移動手段で結ばれた 2 地点間ごとにまとめる。加えて、日常的に持ち歩く携帯端末から得られるデータのみを利用することで、ユーザに負担をかけない移動特性抽出の自動化を実現した。

Automation of Movement Characteristic Extraction for Movement Assistant System

SUZUKI KAZUKI¹ SAKAMOTO DAISUKE² AZUMI TAKUYA³ MOCHIDUKI MASAHIRO⁴ NISHIO NOBUHIKO²

1. はじめに

近年、スマートフォンを始めとした携帯端末の高機能化が急激に進み、様々な種類のセンサが搭載されるようになってきている。携帯端末はユーザが常に持ち歩いているため、携帯端末から得られる位置情報やセンサ情報、無線基地局情報などといった情報が収集可能となっている。本論文ではこのような携帯端末から取得できる情報をライフログと呼ぶ。ライフログを収集し解析することで、個人の行動の特徴を把握し、個人特化したサービスを提供することが可能となる。

個人特化したサービスの 1 つとして、ユーザの移動支援が考えられる。個人特化した移動支援システムの一例に、Google Now [1] がある。Google Now は自宅や職場を自動で認識し、ユーザの望むタイミングで乗り換え案内を提示するなど、個人に適したアドバイスを全自動で行うサービスをすでに実現している。

里中ら [4] は、スマートフォンから得られる GPS 情報と加速度のセンサデータを自動解析し、移動支援を行うシス

テムを提案している。彼らは移動支援に必要な情報を移動特性と定義し、その移動特性を用いることで、様々なユーザの状況に対応可能な移動支援システムの設計・実装を行っている（以下、移動支援に必要な情報を移動特性と呼ぶ）。しかし、彼らの研究では、移動手段を誤認識したり、地下鉄などの対応していない移動手段があったりと正確に移動特性を抽出できていない。

そこで本研究では、移動支援システムのためのユーザの移動特性抽出を自動化することを目的とし、ユーザの歩行速度や移動手段、ユーザが普段利用する経路の特性など、ユーザごとの特徴を加味した移動特性の抽出を行う。本研究での移動特性は、里中ら [4] が定義したものを利用する。最終的な目標として、本研究で抽出した移動特性を利用し、ユーザに特化したサービスを提供可能にすることを目指す。

2. 関連研究

本章では、既存の移動支援システムや、ユーザの状態を認識する研究について記述する。

2.1 既存の移動支援システム

Google Now[1] は、Google が提供している Push 型の移動支援システムである。Google Now はその時々でユーザ

¹ 立命館大学 大学院 情報理工学研究科

² 立命館大学 情報理工学部

³ 大阪大学 大学院 基礎工学研究科

⁴ 立命館大学 総合科学研究機構

にとって有益であると考えられる情報を自動で提示する。たとえば、ユーザの現在位置の天気を提示したり、ユーザが自宅へ帰ると予想されるタイミングを見計らって、自宅への経路案内を提示するなどの支援をしたりというサービスを提供している。しかし、Google Now のサービスはまだ完全に個人特化されているとは言い難い。たとえば、自転車で移動しているユーザに対して、車での移動を前提としたルートを表示してしまうことがある。加えて、バスや自転車のルート案内は一部地域でしか対応していないため、日本全国どこでも均質なサービスを楽しむことができるわけではない。

里中ら [4] は、スマートフォンに搭載されている GPS 情報や加速度センサから得られるデータを解析・利用することでユーザごとに個人特化した移動支援システムを提案している。彼らは、あらかじめ収集した GPS 情報や加速度センサのデータを解析してユーザの滞在や移動特性を抽出しておき、解析したデータと現在のユーザの状況を照らし合わせることで各ユーザに適したアドバイスを提示するシステムの実現を目指している。しかし、この研究では、評価において手で移動特性を抽出しており、センサデータから自動で生成した移動特性を利用してユーザの移動支援を行うには至っていない。

2.2 ユーザの移動状態認識手法

Niken ら [5] は、ユーザが屋内で滞在している場所を Significant Location と名付け、GPS データを用いて Significant Location を抽出する手法を提案している。加えて、Significant Location から別の Significant Location を訪れる際の移動について、歩行区間と歩行以外の区間に区別し、歩行以外の区間における移動手段を最高速度と平均速度を基に、ユーザが移動しているかどうかを判別する手法を提案している。田中ら [6] は、スマートフォンから得られる GPS データについて、位置の分散・速度の算出することで移動・停留の判定を行っている。加えて、ユーザの移動を「歩行・スクーター・自動車・電車・新幹線」の 5 つの状態に分類している。小林ら [7] は、スマートフォンに搭載されている加速度センサ、マイク、GPS を用い、「走行・歩行・自転車・停止・自動車・バス・電車」の 7 つの状態を認識している。

これらの研究では、移動手段の判定に、主に GPS を利用しているため、地下鉄などの GPS が取得できない区間での移動は想定されていない。加えて、「電車に乗っている」、「バスに乗っている」といったユーザがどんな移動状態にあるのかということに主眼を置いており、それらの電車やバスがどこからどこまでの移動であるのかといったことや、同一の移動区間のデータをまとめるといったことについても言及されていない。上記のようなデータを統計としてまとめることは、移動支援をする上で非常に重要であ

る。理由については次章で後述する。

3. 要件

本章では、本研究で移動特性を抽出する際に満たすべき要件について述べる。

3.1 ユーザに負担をかけない形での移動特性の自動抽出

本研究では、なるべくユーザに負担をかけずに移動特性を抽出することを考える。移動特性を抽出するにあたり、特別なセンサなどをユーザの腕や足などに装着させユーザの行動を高い精度で認識させることによって、移動支援システムに必要な移動特性を詳細に抽出できる可能性はある。しかし、常に特別なセンサ類をユーザが装着し続けることは現実的ではない。加えて、移動特性を得るために、ユーザになんらかの情報の入力を強いるなどといった行為も、負担を強いるため好ましくない。普通のユーザの行動になるべく負担や制約をかけずに移動特性を抽出する必要がある。

3.2 移動支援に適用可能な統計情報の作成

毎日の通勤・通学のような、同じ時間に同じ場所を通るような移動においても、必ずしも毎回全く同じ行動を取るとは限らない。また、道路の混み具合などによって移動に要する所要時間も毎回変化する。そこで本研究では、1 回 1 回のユーザの移動を認識するだけでなく、統計としてまとめることで、移動支援に有益なデータを抽出する。

3.3 場所の制約を受けない移動特性の抽出

2.1 節で述べたシステムのように、ユーザの移動特性を把握する上で地図情報などのデータを利用すると、ユーザの移動を正常に認識できない場所が存在するため好ましくない。そこで本研究では、場所の制約を受けないユーザの移動特性を認識することを目指す。

4. 移動特性の抽出

本章では、本研究で抽出する移動特性の定義と、その抽出方法について詳しく記述する。まず、4.1 節では本研究で抽出する移動特性に関して述べる。移動特性の具体的な抽出方法については、4.2 節以降で述べる。

4.1 移動特性の定義と用語説明

移動特性は以下に説明する立ち寄りポイントおよび立ち寄りポイント間の移動 (2 点間移動)、乗り換えポイント、チェックポイントのことを指す。

4.1.1 立ち寄りポイントと 2 点間移動

里中らはまず、ユーザの移動を「屋内の停滞場所」と「移動」に区別している。屋内の停滞場所とは、具体的には自宅や本屋、大学の図書館など、ユーザが一定時間以上立ち

寄っている場所を指し、それを立ち寄りポイントと呼んでいる。そして、立ち寄りポイント間での移動を2点間移動と名づけている。

4.1.2 乗り換えポイント

次に、里中らは2点間移動において、さらに細かくユーザの移動を分類している。里中らはまず、ユーザの移動を駅やバス停、駐輪場といった移動手段が変化するポイントに着目し、これらのポイントを乗り換えポイントと定義している。そして、ユーザの移動は乗り換えポイント間の移動手段の組み合わせで表現できることを言及している。加えて、乗り換えポイント間の移動ごとに統計データをまとめることで、様々な状況に対応することを考えている。たとえば、ユーザが自宅から初めての場所に訪れる際に移動支援をする場合でも、今まで利用したことのある乗り換えポイントまでの移動を部分的に利用することで、より適用可能な範囲を増やした移動支援をする手法を提案している。

4.1.3 乗り換えポイント間の移動特性

次に、乗り換えポイント間の移動特性について説明する。始点と終点の乗り換えポイントが同一でも、複数の移動ルートが混在するケースが存在する。例えば、同じ駅と大学間のバス移動で経由地点が異なる場合といったことである。里中らは、このように複数のルートがあった場合には、複数のルートに合わせた移動支援を行うために別々の統計としてまとめるべきであると主張している。さらに、同一のルートにおいても信号やバス停などによる停滞が毎回異なることによって、所要時間が大きく変化することから、信号やバス停などによる停滞のデータも統計として残しておく必要性を述べている。里中らは、信号やバス停などによる、移動中にユーザが停滞するポイントをチェックポイント候補と呼び、その中でも特にユーザの移動に影響を与えるポイントをチェックポイントと呼んでいる。

4.2 移動特性抽出手法

移動特性を抽出するアルゴリズムは、大きく分けて「同一の2点間移動の抽出」、「2点間移動における乗り換えポイントの抽出」、「同一の乗り換えポイント間移動の抽出」という3つのフェーズから構成されている。

4.3 同一の2点間移動区間の抽出

本項では、「同一の2点間移動区間の抽出」について説明する。「同一の2点間移動区間の抽出」アルゴリズムの概要を図1に示す。スマートフォンから得られるセンサデータから立ち寄りポイントを生成する。立ち寄りポイントの抽出アルゴリズムは、里中ら [4] が利用した手法と同様のものを利用する。その際に、次に訪れた立ち寄りポイントの情報を記録しておく(図中Aの処理)。次に、それらの立ち寄りポイントごとの遷移を2点間移動に分割する(図中Bの処理)。この処理を再帰的に繰り返すことで、実際

の移動に対応するような2点間移動のログを生成する。その後同一となる2点間ログごとにまとめることで、「同一の2点間移動」を抽出する。

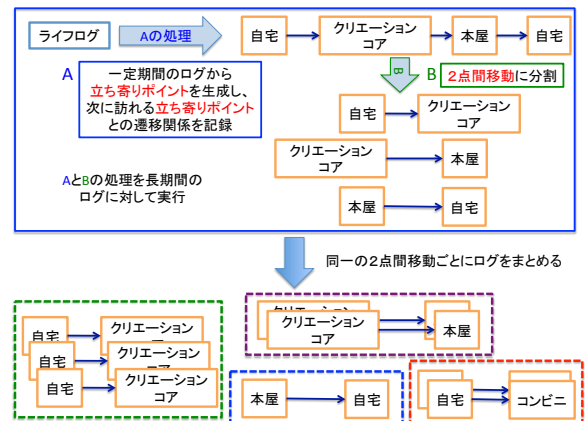


図1 同一の2点間移動区間の抽出

4.4 2点間移動における乗り換えポイントの抽出

続いて、同一の2点間移動についてそれぞれ解析する。抽出アルゴリズムの概要を図2に示す(図中クリエーションコアは大学構内の建物)。解析は大きく5つのフェーズに分かれている。

4.4.1 歩行区間の検出

はじめにユーザの歩行区間に着目する。青木ら [8] は、ユーザの移動手段が変化する際には、必ず歩行区間があることを言及している。ユーザの歩行区間を検出することで、ユーザの移動区間を歩行区間とそれ以外の区間に区切る。そうすることで、乗り換えポイントは、ユーザの歩行区間の切れ目のいずれかとなる。

- 携帯端末の加速度センサを利用したステップの取得
ステップの取得にはスマートフォンの加速度センサを利用した歩数計アルゴリズムを用いる。歩数計アルゴリズムについては、今回、吉見ら [9] が開発したアルゴリズムを利用した。
- 歩行区間検出アルゴリズム

吉見らの歩数計アルゴリズムは、加速度センサの振幅の大きさを利用してステップを検出している。そのため、たとえば端末を大きく振るなど、大きな振動が起こればステップを誤検知してしまうため、必ずしも正確にステップが観測できるとは限らない。図3は、あるユーザにおいて、実際の移動中におけるユーザのステップを可視化したものである。横軸は時刻であり、ステップを検出した場合、対応する時刻に青線が引かれるようになっている。図3を見ると、バスや電車の区間においてもステップが観測されていることがわかる。

一方で、徒歩区間において、図3の左図を見ると、一

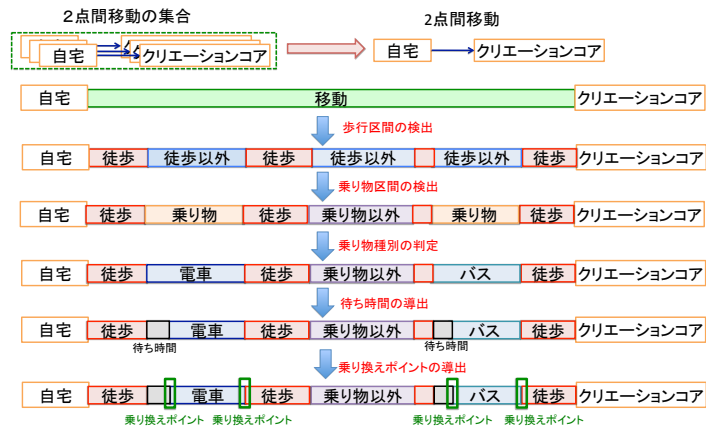


図 2 2 点間移動における乗り換えポイントの抽出

見、常に連続してステップが観測されているように見える。(図中 A) しかし、図中 A の区間を拡大した図(図 3 右図)を見るとわかるように、実際にはまばらな部分が存在する。

そこで、一定時間間隔以内で連続してステップが観測されている区間を歩行区間とみなす。この際、1~2 秒程度の短い間隔のみステップが取れないような場合(図 3 右図のまばらな部分)にも、歩行区間が続いているとみなす。加えて、連続したステップが一定時間に満たない場合には、その歩行はノイズであるとみなし歩行区間から除外する。

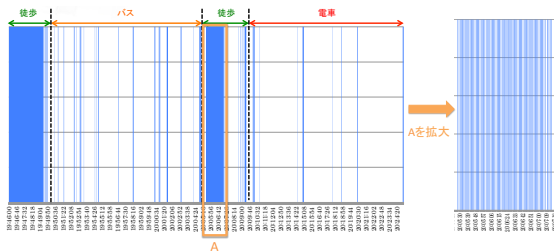


図 3 実行動におけるユーザのステップ記録

● リズミカルステップによるノイズ除去

上記したように、一定時間に満たないステップ区間を除去することで、ノイズを除去することが可能である。しかし、加速度センサでステップを検出しているため、例えば、たまたまバスに乗車中、連続してステップが取れてしまい、歩行として誤検知されるというケースも考えられる。そこで、一定時間に観測できるステップの数の違いに着目する。図 4 は、ユーザのステップが観測された時刻について、歩行時とバス乗車時を比較したものである。図 4 に歩行時とバス乗車時のステップログを示す。バス区間でのステップ(ノイズ)は、1 秒間に 1 回のみステップが観測される場合が多いのに対して、実際の歩行区間では、1 秒間に 2~3 回のステップが観測されている。

このような、実際の歩行の特性を利用し、一定時間内に一定歩数以上のステップをリズムカルステップと呼び、このリズムカルステップのみを真の歩行とみなす。そして、リズムカルステップでないステップはノイズとして除去する。図 6 は、1 秒間に 2 歩以上歩行している場合のみをリズムカルステップとみなして、リズムカルステップのみをステップとして可視化したものである。単純にステップが検知された時刻を可視化した図(図 5)と比べると、バス乗車時や電車乗車時のノイズが軽減していることがわかる。

歩行区間におけるステップログ

- 2013-07-02 19:46:00
- 2013-07-02 19:46:00
- 2013-07-02 19:46:01
- 2013-07-02 19:46:01
- 2013-07-02 19:46:02
- 2013-07-02 19:46:02
- 2013-07-02 19:46:03
- 2013-07-02 19:46:03
- 2013-07-02 19:46:04
- 2013-07-02 19:46:04
- 2013-07-02 19:46:05
- 2013-07-02 19:46:05
- 2013-07-02 19:46:06
- 2013-07-02 19:46:06
- 2013-07-02 19:46:07
- 2013-07-02 19:46:07
- 2013-07-02 19:46:07

1秒間に1ステップではなく、
1秒間に2~3ステップ歩いている

バス区間におけるステップログ

- 2013-07-02 20:00:24
- 2013-07-02 20:00:25
- 2013-07-02 20:00:27
- 2013-07-02 20:00:46
- 2013-07-02 20:00:50
- 2013-07-02 20:00:52
- 2013-07-02 20:00:54
- 2013-07-02 20:00:57
- 2013-07-02 20:01:36
- 2013-07-02 20:01:37
- 2013-07-02 20:01:38
- 2013-07-02 20:01:39
- 2013-07-02 20:01:41
- 2013-07-02 20:01:41
- 2013-07-02 20:01:42
- 2013-07-02 20:02:58
- 2013-07-02 20:02:59

1秒間に1ステップ
観測されている場合が多い

図 4 歩行時とバス乗車時で観測されるステップの違い

4.4.2 乗り物区間の検出

2 点間移動について、4.4.1 項でのアルゴリズムを適用すると、2 点間移動は、図 7 のように区切ることが可能となる。次は歩行以外の区間について、その区間が乗り物に乗車中か否かを判定する。図 8 に判別方法のフローを示し、詳しい手法については、以下に詳しく述べる。

● 距離による判定

歩行以外の区間を判別するにあたり、まず 1 つの区間における出発地点と到着地点の距離を用いて、乗り物

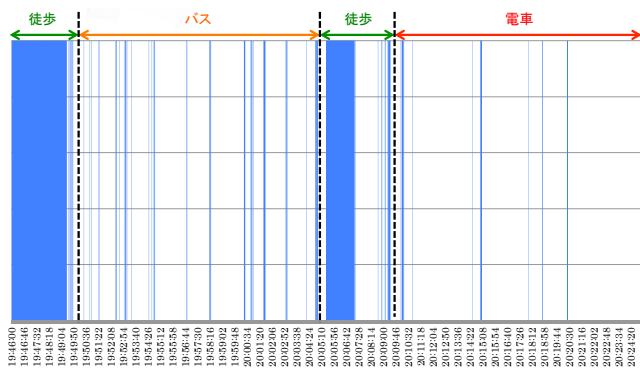


図 5 リズミカルステップ適用前

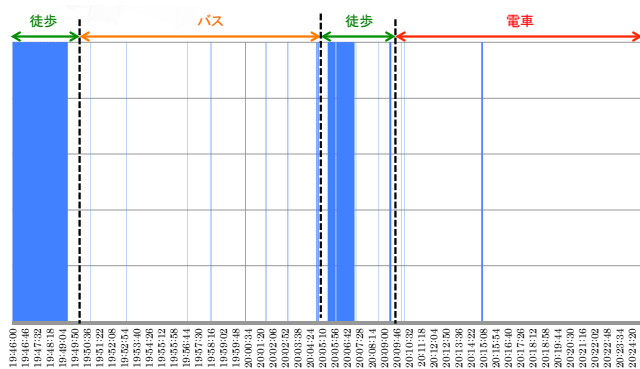


図 6 リズミカルステップ適用後

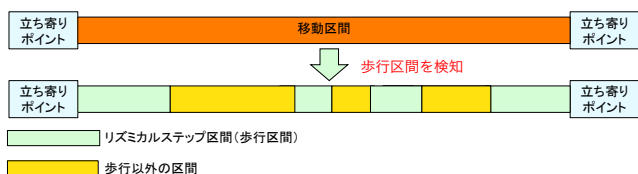


図 7 歩行区間の検出

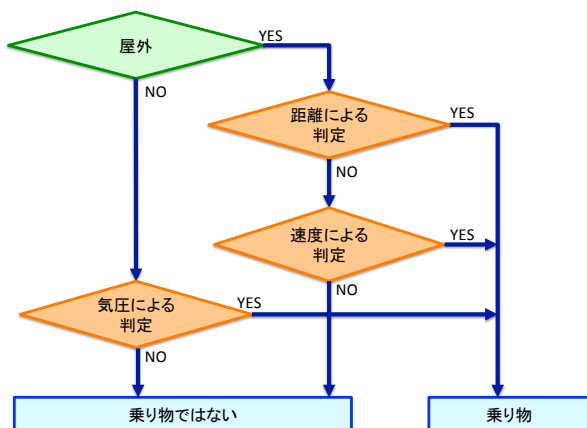


図 8 乗り物判別フロー

であるか否かを判定する。4.4.1 項でも述べたように、ユーザが乗り換えを行う際には、歩行区間が必ずあるため、1つの歩行以外の区間に複数の乗り物は混在しない。そのため、1つの区間における移動開始地点と移動終了地点が一定距離以上離れていれば、その区間は乗り物に乗って移動していたと考えられる。

上記の距離は、移動開始地点と移動終了地点付近で観測された GPS データにヒュベニの公式を適用することで算出することが可能である。しかし、ユーザが屋内にいる場合には GPS が観測できないため、地下鉄のような屋内移動にはこの手法は使えない。そこで GPS が観測できなかった場合には、後述する気圧による判定を利用する。

● 速度による判定

次に速度を用いて乗り物判定を行う。屋外における移動の場合、移動開始地点と移動終了地点の距離を用いることで、屋外の乗り物区間を検知することが可能である。しかし、乗り物を用いて移動する距離は、ユーザの状況によって大きく異なるため、一定以上の距離ならば、乗り物による移動であると明確に定義することは難しい。そこで速度によって乗り物か否かを判定することを考える。GPS を用いて速度を算出し、速度が一定以上である場合に乗り物による移動であると判定する。速度の算出には、松倉ら [10] が提案した手法を用いた。本研究では区間中において、上記手法にて算出した一秒間での最高速度を認識し、その速度が一定以上である場合に乗り物とみなす。しかし、GPS が観測できない場合には速度が算出できないため、この手法も地下鉄などの屋内移動においては適用できない。

● 気圧による判定

最後に気圧センサの値を利用して乗り物判定を行う。乗り物乗車時にも気圧は大きな変化を及ぼす。実際に、渡邊ら [11] は地下鉄乗車時に、米田 [12] は電車がトンネルを通過時に、気圧が大きく変動することについて言及している。

本研究では、地下鉄の移動判定に気圧センサを利用する。これは、渡邊らが言及しているように、地下鉄乗車時には気圧が大きく変動するため、気圧センサの変化が地下鉄判定に利用できると考えたためである。なお、気圧データの変化の検出には、米田 [12] が実装した検知機構を用い、気圧変動のしきい値を、地下鉄判定を行うのに最適なしきい値に変更した。

4.4.3 乗り物種別の判定

次に具体的な乗り物の種別の判定を行う。本研究では速度と GPS 情報を基に乗り物の種別を判定する。上記で乗り物と判定された区間のうち、最高速度が 70km/h 以上である移動手段を電車、25~70km/h を自動車・バス、25km/h 以下を自転車とし、出発地点から到着地点まで一度も GPS

が取れていない区間の乗り物を地下鉄とした。

4.4.4 待ち時間の導出

電車や地下鉄などの乗り物区間を正確に算出するためには、乗り物が発車した時刻と到着した時刻で乗り物区間を正確に区切る必要がある。ユーザは徒歩で乗り物に乗り込むため、ステップは観測される。しかし、ユーザが乗り物に乗る際の歩行はごく短時間であるため、4.4.1項で示した手法を適用するとノイズと誤検知されてしまうケースが多い。加えて、電車の待ち行列を進むようなゆっくりとした移動区間においてはリズムカルステップと判定されることが多い。実際に、図5、6を見ると、徒歩からバスに乗り込む際に観測されているステップが除去されている様子が見て取れる。そのため、4.4.1項で示した手法を適用すると、待ち時間も含めて乗り物区間と認識されてしまうことが多い。一方で、乗り物に乗る際のステップを観測するようにノイズ除去の閾値を下げると、余分なノイズまで観測されてしまう問題が発生する。

そこで、乗り物区間においてのみ、ステップによる区間判定と乗り物判定を再帰的に行うアルゴリズムを提案する。アルゴリズムの概要を図9に示す。前述したように、ノイズとして省かれるだけで、乗り物に乗った瞬間のステップの観測は可能なため、ステップを用いて乗車を判定すること自体は可能であると考えられる。ステップが観測された時刻を乗り物区間の切れ目の候補とし、その切れ目以前の区間において再度乗り物判定を行う。仮に、ステップが観測された時刻が実際の乗車時であった場合は、その区間以前が乗り物とは判定されないため、乗り物と判定されなくなるまで再帰的に処理を行い続ける。

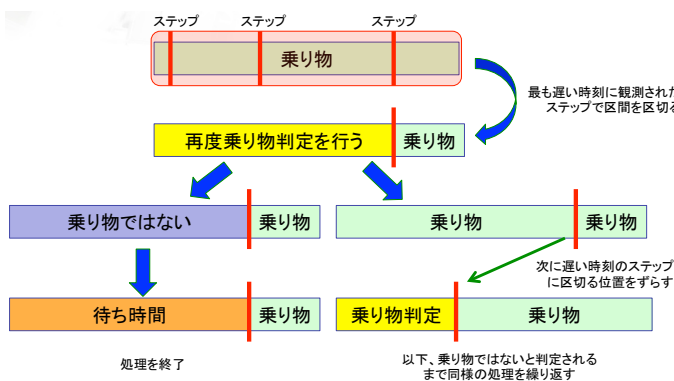


図9 待ち時間判定アルゴリズムの概要

4.4.5 乗り換えポイントの導出

最後に乗り換えポイントの導出を行う。4.4.1項から4.4.4項で示したアルゴリズムを用いて、区切られた区間のうち、乗り物→徒歩、徒歩→乗り物に移動手段が変化した時刻（以下乗り換え時刻と呼ぶ）とGPS情報を基に乗り換えポイントを生成する。しかし、乗り換えポイントが屋内にある場合、乗り換え時刻にGPSが観測できない。そ

で、乗り換えポイントが屋内にある場合には、GPSの消失点・復活点を利用して乗り換えポイントを作成する。

● 屋外における乗り換えポイントの生成方法

乗り換えポイントが屋外に存在する場合には、乗り換え時刻から最も近い時刻で取得したGPSの位置情報を利用する。この際に、直近で観測されたGPSの緯度経度を中心として、1辺の長さがGPSのAccuracyの2倍となるような矩形を生成する。ただしGPSのAccuracyは、数10mを超えることがあるため、そのような場合に1辺の長さをAccuracyの2倍とすると、必要以上に矩形が大きくなってしまふ。そこで本研究では、Accuracyが10mを超えた場合には、Accuracyを10mとして矩形を算出した。

● 屋内における乗り換えポイントの生成方法

乗り換えポイントが屋内に存在する場合には、乗り換え時刻においてGPSを観測することができない。しかし、同一の乗り換えポイント間の移動ごとに移動ログをまとめたいため、乗り換えポイントが屋内にある場合でも、乗り換えポイントを作成する必要がある。そこで、地下鉄の乗り換えポイントは、地下鉄に乗る際には、乗り換え時刻に対して直前のGPSが観測された地点（GPSの消失点）、地下鉄から降りる際は乗り換え時刻直後でGPSが観測された地点（GPSの復活点）を代理の乗り換えポイントとして生成を行う。ただし、これらのGPS情報は、あくまで乗り換えポイント間の移動をまとめるために利用するだけであり、乗り換え時刻については4.4節で算出した時刻を用いる。

4.5 同一の乗り換えポイント間移動の抽出

全ての2点間移動における解析が終わったら、次はそれらのデータを同一の乗り換えポイントごとに分割し、始点と終点が同一の乗り換えポイントごとにまとめる。この際に逆向きの移動は別の区間としてまとめる。

4.4.5項で生成した乗り換えポイントの位置情報を利用して、同一の乗り換えポイント間移動を抽出する。以下に抽出アルゴリズムを示す。

同一の乗り換えポイント間移動を判定するためには、始点と終点の乗り換えポイントが一致していればよい。そのためにはまず、乗り換えポイント同士が一致しているかを判定する必要がある。単純に考えると、距離が近い乗り換えポイントをまとめれば良いように思える。しかし、乗り換えポイントが隣接している場所では、単純に距離だけで判定すると、別の乗り換えポイントと誤認識してしまう恐れがある。そこで、乗り物区間におけるもう一方の乗り換えポイントを利用する。

判定アルゴリズムの概要を図10に示す。図10は、南草津駅を例とした同一の乗り換えポイントと判断する場合のアルゴリズムを図示したものである。判定したい乗り換え

ポイントの距離を比較する(図中(1)). 次に乗り物区間におけるもう一方の乗り換えポイントについても距離を比較し、一定距離内にあるかを調べる(図中(2)). 両方を満たした場合に判定したい乗り換えポイントを同一の乗り換えポイントとして識別する.

一方で、南草津駅バス停の場合は、もう片方の乗り物区間が京都駅ではなく、別の場所とつながっているため、この手法を用いることで、たとえ南草津駅と南草津駅バス停のように距離が近い場合でも間違った方の乗り換えポイントにまとめてしまうことを防ぐ。しかしこの手法で南草津駅を判定したとき、大阪駅-南草津駅、京都駅-南草津駅のように、もう一方の乗り換えポイントが異なる場合には南草津駅が同一の場所とは判定されなくなる。そこで移動手段が同一であり、かつ判定したい乗り換えポイントが一定距離以内にある場合にも同じポイントとしてまとめた。

なお、乗り換えポイントが同一と判定された場合には、2つの乗り換えポイントにおける最大緯度・最大経度、最低緯度・最低経度を矩形の頂点として、あらたに矩形を作り直す。

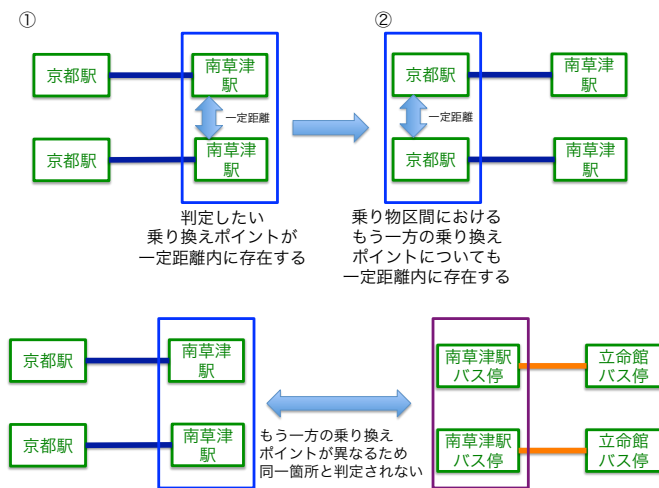


図 10 乗り換えポイントの同一性判定

5. 評価

本研究ではプロトタイプとして、4.3 節、4.4 節、および、4.5 節の内容について実装を行った。評価項目として、乗り換えポイントが正確に抽出できているかについて評価する。

評価のため被験者 1 名に端末を携帯してもらい、14 日間、ライフログを収集した。加えて乗り物の乗降車時に、移動手段、場所、時刻を手動で記録してもらった(これを記録乗り換えポイントと呼ぶ)。収集したライフログから乗り換えポイントを自動生成した。自動生成の際には、時刻、ポイントの緯度経度、移動手段(車・電車・自転車・地下鉄)が生成される。この自動生成した乗り換えポイント

が記録乗り換えポイントの移動手段と場所が合致している上で、生成された時刻と、記録した時刻の差が 1 分以内なら正解として、適合率、再現率を算出した。それぞれの算出式を式 1、式 2 に示す。

$$\text{適合率} = \frac{\text{正解数}}{\text{生成された乗り換えポイント数}} \quad (1)$$

$$\text{再現率} = \frac{\text{正解数}}{\text{記録乗り換えポイント数}} \quad (2)$$

評価結果を表 1 と表 2 に示す。

表 1 全体の評価結果

	合計
生成乗り換えポイント	113
記録乗り換えポイント	132
正解数	69
適合率	0.611
再現率	0.523

表 2 個別の評価結果

	車・バス	電車	自転車	地下鉄
生成乗り換えポイント	27	34	46	4
記録乗り換えポイント	44	40	44	4
正解数	20	24	22	3
適合率	0.741	0.706	0.478	0.750
再現率	0.455	0.600	0.500	0.750

全体の適合率は 0.61、再現率は 0.52 という評価結果が得られた。個別の評価結果をもとに乗り換えポイントの抽出を失敗した原因を考察する。

車・バスの抽出が失敗した原因としては、速度超過による電車との誤認識である。これは乗り物種別の判定を速度を元に行っているためと考えられる。被験者の車での移動の際に、最高速度が一時的に 70km/h を超過したため、電車と誤認識される事があった。自動車の一般道における制限速度は、60km/h であることから、車と電車を区別するためのしきい値を今回 70km/h としたが、これを一時的に超過したことがあったため、電車と誤認識された。単純に速度をしきい値とした場合、高速道路における移動などでも誤認識が起りうる問題であるため、今後これに対応する必要がある。

電車の抽出が失敗した原因としては、電車乗り換えの一部が抽出されていなかったためである。被験者の電車移動の中に、駅内での路線乗り換えの経路が存在していた。乗り換え先の電車は発車時刻までの間、プラットフォームで待機している電車であり、発車までの間、電車内で待機する必要はある。被験者が電車内で待機をすると、歩行が止まるためその時点最後にステップが検出されなくなる。そのため、この待機開始時間を電車の発車時間であると誤認してしまい、抽出失敗となってしまった。

自転車の抽出が失敗した原因としては、歩行区間の切り分けが正確に行えていなかったためである。本研究では、加速度センサを利用して歩数を計測し、歩行区間の切り分けを行っている。そのため、自転車をこいだ際の動きも歩行と判定され、間違った乗り換えポイントの生成地点が行われてしまったと考えられる。

地下鉄の抽出を失敗した原因としては、失敗した1回において、待ち時間判定が正しく行えていなかったためである。歩行区間の切り分けが正しく行えなかったために、発車した時刻よりも前の時刻に歩行区間が切り分けられてしまい、時刻差が今回正解のしきい値とした時刻差に収まらなかった。

6. おわりに

本研究では、既存の移動支援システムや移動認識における研究の問題点を挙げ、それらの問題を踏まえた上で移動支援に利用可能な移動特性に関する要件を定義した。スマートフォンに搭載されているセンサ類を用いて、ユーザの移動支援に必要な移動特性の抽出の自動化を行った。その上で評価を行い、半分程度の乗り換えポイントの抽出に成功した。しかし精度としては十分であるとは言いがたく、今回判明した問題点について今後考慮していく必要がある。加えて既に得られた移動特性データを利用することで移動特性抽出の精度をさらに良くしたり、機器不良などで乗り換えポイントが欠損していても補完したりする必要があると考える。

参考文献

- [1] Google now, (Online) 2014-2-05. <http://www.google.com/landing/now/>.
- [2] Google マップ, (Online) 2014-2-05. <https://maps.google.co.jp/>.
- [3] Siri, (Online) 2014-2-05. <http://www.apple.com/jp/ios/siri/>.
- [4] Yusuke Satonaka, Takumi Kitazawa, Kazuki Suzuki, Yuki Fukuzaki, Takuya Azumi, and Nobuhiko Nishio: LifeLog-Based Active Movement Assistant System, 1st IEEE International Conference on CyberPhysical Systems, Networks, and Applications(CPSNA2013), 2013.
- [5] Niken Tri Mahayani, Takumi Kitazawa, and Nobuhiko Nishio. Mining life-log sensing data to extract user's significant locations and movement (アドホックネットワーク). 電子情報通信学会技術研究報告: 信学技報, Vol. 112, No. 30, pp. 117-121, may 2012.
- [6] 田中剛, 鈴木誠二, 土井千章, 中川智尋, 稲村浩, 太田賢, 峰野博史. ライフログ生成のための屋外状況推定手法の検討. マルチメディア、分散、協調とモバイル (DICOMO2013) シンポジウム, pp. 1057-1064, jul 2013.
- [7] 小林亜令, 岩本健嗣, 西山智. 釈迦: 携帯電話を用いたユーザ移動状態推定・共有方式 (モバイルコンピューティング, モバイルアプリケーション, ユビキタス通信, モバイルマルチメディア通信). 電子情報通信学会技術研究報告. MoMuC, モバイルマルチメディア通信, Vol. 108, No. 44, pp. 115-120, may 2008.
- [8] 青木政勝, 瀬古俊一, 西野正彬, 山田智広, 武藤伸洋, 阿部匡伸. GPS 未計測区間における移動手段判定手法の検討. 情報処理学会研究報告. UBI, No. 110, pp. 39-44, nov 2008.
- [9] 吉見駿, 新田知之, 安積卓也, 西尾信彦. PDR を活用した地下街ナビゲーション進捗適応機構. マルチメディア、分散、協調とモバイル (DICOMO2013) シンポジウム, pp. 1039-1050, jul 2013.
- [10] 松倉祐, 光山哲平, 宮崎孝信, 里中裕輔, 新井イスマイル, 西尾信彦. ライフログセンシングにおける停留判定精度向上のための GPS 測位誤差除去手法. マルチメディア、分散、協調とモバイル (DICOMO2010) シンポジウム, pp. 1418-1425, jul 2010.
- [11] 渡邊孝文, 上坂大輔, 村松茂樹, 小林亜令, 横山浩之. 気圧センサを利用した昇降状態を含む移動状態推定, 電子情報通信学会技術研究報告. MoMuC, モバイルマルチメディア通信, Vol. 111, No. 296, pp. 19-23, nov 2011.
- [12] 米田圭佑, 望月祐洋, 西尾信彦. 気圧センシングを用いた行動認識手法. 情報処理学会研究報告. UBI, [ユビキタスコンピューティングシステム], Vol. 2014, No. 14, pp. 1-8, mar 2014.