

# PDR 推定軌跡の簡略化による経路理解の向上

坂東 秀人<sup>1</sup> 松浦 大希<sup>1</sup> 西尾 信彦<sup>1</sup>

**概要:** 近年, 屋内にいるユーザの位置情報を知ることはナビゲーションだけでなくマーケティングにおいても非常に重要な情報であり, 注目を集めている. しかし, 屋内でユーザの位置情報を推定する手法は初期整備が必要なものが多く, 初期整備の必要のない PDR はその測位手法の特徴から測位誤差が累積する問題がある. PDR による測位結果を補正する手段として, 他の測位手法を併用するものや, マップマッチングのような事前に用意した建物情報を用いるものがあるが, どちらも導入コストが増加してしまう. 本研究では PDR によって推定された移動軌跡を線分化, および直交化を行うことで簡略化するシステムを提案する. 建物内に存在する通路は互いに直交している場合が多く, そこを歩行するユーザの移動軌跡も同じく直交したものとなる. すなわち, マップ情報を用意せずとも移動軌跡の補正を行うことが可能である. 提案手法により, 推定軌跡を簡略化することで累積誤差を減らし, ユーザの移動経路を明確にする. 評価実験として, 店舗内を歩行するユーザの歩行軌跡の簡略化を行い, 軌跡の補正と誤差の計測を行なった. 結果として, 補正された軌跡は簡素で移動軌跡が明確なものとなり, 約 58 % の誤差低減を確認した. しかし, 実験時に一回転するようなイレギュラーな歩行が見られ, その動作をプログラムでは正確に認識し処理することができず, 誤った進行方向へと補正されてしまった.

SYUTO BANDO<sup>1</sup> TAIKI MATSUURA<sup>1</sup> NOBUHIKO NISHIO<sup>1</sup>

## 1. はじめに

近年, 屋内にいるユーザの位置情報を知ることはナビゲーションだけでなくマーケティングにおいても非常に重要な情報であり, 注目を集めている. 屋内のユーザの位置情報を知る手法として建物内に設置された Wi-Fi や Bluetooth Low Energy(BLE) を利用するもの [1][2] や, ユーザが身につけたセンサ情報を用いる歩行者自律測位 (PDR: Pedestrian dead-reckoning)[3] がある. 特に PDR は環境側の整備が必要なく, 導入コストを抑えられる利点がある. しかし, PDR はあくまで測位開始地点からの相対的な位置を求める測位であるため, 測位時間の増加とともに測位誤差も増加するという問題がある. そこで PDR を行う際には他の測位手法の併用 [4] や, マップマッチングを利用することがある [5][6]. マップマッチングとは, 事前に建物内の歩行可能な通路を線 (リンク) とその交点 (ノード) で表現した歩行空間ネットワークを作成しておき, PDR によって推定された移動軌跡 (以下, 推定軌跡) をその線上へと補正を行うものである. しかし, マップマッチングを行うためには詳細な歩行空間ネットワークが必須となるため, 他の測位手法と同様に導入コストが高くなってしまふ. さらに, 人間の歩行には左右のブレが含まれるため, PDR の推定軌跡

は波打ったような線として描かれる. そのような推定軌跡ではどの部分で直進しているのか, どの部分でどちらの方向へ曲がったのか, という情報を判断することが困難であり, またマップマッチング等のプログラムで処理する際にも処理が複雑になってしまう. そこで本研究では, PDR によって得られた推定軌跡を歩行空間ネットワークを用いることなく, 簡略化を行う. 簡略化とは推定軌跡を簡素なものとするすることで, ユーザの移動経路を明確にするものである. 具体的な簡略化の処理として, 線分化と直交化の 2 つを行う. 1 つ目の線分化とは, 直進していると判断できる部分を 1 つの線分に変換し, 推定軌跡を連続した線分の集合として表現することである. 2 つ目の直交化とは, ユーザの進行方向変化を 90 度単位の変化として扱うことである. 建造物内の通路, 特に商品棚が多く並ぶ店舗内の通路は, それぞれが平行または直交している場合が多く, その通路を歩行するユーザの移動軌跡もそれに準じたものとなる. そのため, 推定軌跡を通路と同じく線分それぞれが直交したと形へと変換することで, 誤推定や誤差によって通路から大きく外れた軌跡を減らすことができると考える.

## 2. 関連研究

### 2.1 屋内測位手法

屋内でユーザの位置情報を測位する手法を紹介する.

<sup>1</sup> 立命館大学 情報理工学部

### 2.1.1 Wi-Fi 基地局を用いた測位

Alfakih ら [1] は, Wi-Fi 基地局から発せられる電波の信号強度を混合ガウスモデルを用いたマップを作成し, 測位 (以下, Wi-Fi 測位) に利用している. Wi-Fi 測位はあらかじめ建物内に設置されている基地局を利用する. そのため初期機材設置の必要はないが基地局の移動や消失に対応する必要があり, 定期的なマップ情報の更新が必要である. また基地局情報が既知である必要があるため, マップを作成した場所でのみ利用可能である.

### 2.1.2 BLE を用いた測位

Ozer ら [2] は, 建物内の各所に設置された BLE ビーコンを利用し, 電波の信号強度からカルマンフィルタを用いてセンサノイズを軽減することでユーザの移動軌跡を推定する (以下, BLE 測位). BLE 測位は初期に BLE ビーコンを設置する必要があるが, 低消費電力で通信が可能であるという利点がある. しかし, 電波が届く範囲が 20, 30m 程度と短く, 無線信号がノイズ等の電波減衰の影響を受けやすいという問題点がある.

### 2.1.3 PDR

PDR は加速度センサ, 角速度センサなどからユーザの一步分の歩行を検知し, 推定される進行方向へ推定軌跡を一步分進ませることで初期地点からの相対的な移動軌跡を求めるものである. Kouroggi ら [3] は, スマートフォンのセンサを利用し, センサが様々な保持状態であっても PDR を行う手法を提案している. PDR はマップ情報を用いないことから未知の場所での測位が可能であるが, あくまで相対的な移動軌跡を推定するものであり, 時間経過によって蓄積したセンサ誤差に過分に影響を受ける.

## 2.2 PDR 推定軌跡補正に関する研究

### 2.2.1 複数の測位手法を組み合わせる手法

Miyazaki ら [4] は, PDR と Wi-Fi 測位を組み合わせることで, それぞれの測位結果を補正し合う手法を取っている. PDR で必要な初期位置, 初期進行方向の設定を Wi-Fi 測位によって推定することが可能であり, PDR の測位精度を Wi-Fi 測位によって評価している.

### 2.2.2 カルマンフィルタを用いた角速度センサ誤差低減手法

田川ら [5] はカルマンフィルタとマップマッチングを用いることで角速度センサの誤差低減を行なっている. センサから得られる角速度を積分し, カルマンフィルタを用いて誤差を低減している. カルマンフィルタだけでは経過時間による累積誤差は低減することができないため, その補正としてマップマッチングを行い, マッチング先のリンクの向きと進行方向を一致させることで累積誤差の除去を行なっている.

### 2.2.3 歩行空間制約を用いた補正

吉見ら [6] は移動軌跡の特徴がマップマッチングによっ

て失われることを危惧し, 推定軌跡が歩行可能な空間外に出た際に補正を行なっている. 通路の壁を歩行空間制約として用いて歩行可能な空間区別することで, 推定軌跡が空間内に収まるかどうかを検知している. ユーザの推定軌跡が壁と衝突し歩行可能な空間から外れたと判定されるごとに, 空間内に収まる歩幅と進行方向を計算し, 推定軌跡の補正を行なっている.

## 3. 提案手法

### 3.1 システムの概要

本節ではシステム全体の概要について述べる. 推定軌跡を簡略化するためには簡略化可能な部分とそうでない部分を区別し, それぞれパーツ分けする必要がある.

#### 3.1.1 xy 座標平面への変換

はじめに, 推定軌跡を計算処理の容易な xy 座標平面上の座標点に変換する. PDR による位置推定は一步進んだと判断されるたびにその地点の緯度経度を推定し, 推定軌跡はそれらの点の連続したものとして表現される. 測位開始地点を xy 座標平面の原点  $(x, y) = (0, 0)$ , 単位をメートル (m), 東方向を x 軸の正の方向, 北方向を y 軸の正の方向として点群の座標変換を行う. 同時に座標点同士の位置関係から進行方向を計算し, 図 1 のように前地点の進行方向と次点への位置関係から角度差を同時に記録しておく. このとき前地点の進行方向に対して右方向を正, 左方向を負とする. これにより, 現在の進行方向は前地点の進行方向に対して「右方向へ 5 度 (+5 度)」「左方向へ 3 度 (-3 度)」と表現できる.

#### 3.1.2 座標点の分類

次にパーツ分けするために点群にラベルづけを行う. ラベルづけの判定基準には第 3.1.1 章で求めた角度差を利用する. 知見によって得られたパラメータを元に閾値を設定し, 5 歩間の角度差の合計値が閾値以上であるかどうかによって点群を「直進」「曲進」「undefined」のいずれかに分類する. ここで「直進」とはユーザがまっすぐ進むもの, 「曲進」とはユーザが左右いずれかの方向へ進行方向を変えたもの, 「undefined」とは「曲進」の中でも特に大幅に進行方向を変えたもので, 進行方向が 5 歩間で 180 度近く変わったものとする. 図 2 にそれぞれのパターンのイメージを示す.

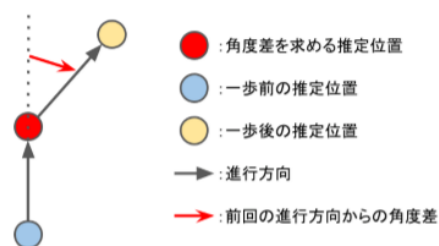


図 1 記録しておく角度差

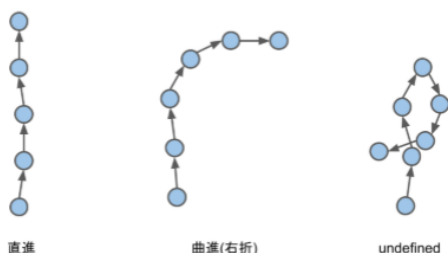


図 2 分類するパターン

### 3.1.3 undefined パーツの扱い

第 3.1.2 章の分類によって「undefined」ラベルがつけられた点 (以下, undefined 点) は, 進行方向が急激に変化した点であるため, ユーザが通路を途中で引き返し, 進行方向を 180 度変更した点であると考えることができる。しかし, そうでない場合であっても同じく undefined 点となっていることを考慮する必要がある。店舗内においてユーザは買い物客である場合がほとんどであり, 買い物客は通常の歩行者と比較して歩行状態が非常に変則的である。買い物客は通路を進む歩行のほか, 周囲を見渡す動作, 棚からの後ずさり, 他の顧客や棚を避ける動作, などといった通常時の歩行にはほとんど見られない動作を過分に行う。それらの動作を行なった場合にも undefined 点, またはそれが連続した点群 (以下, undefined パーツ) と判断されてしまう場合がある。また, 同じ U ターンであっても, 一つの通路内での U ターンや通路をまたぐ U ターンである場合がある。すなわち, 通路情報に大きく左右されてしまうため, 判断を行うことが非常に困難である。また, undefined パーツは進行方向変化を過分に含むパーツであるため, 安易に簡略化を行うことは推定軌跡補正に多大な影響を及ぼす可能性がある。そのため, 本研究では undefined 点と判断された区間は簡略化を行わず, 推定軌跡をそのまま保持することとした。

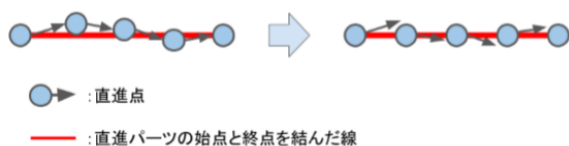


図 3 直進パーツの補正

### 3.1.4 直進パーツの簡略化

第 3.1.2 章の分類によって「直進」ラベルがつけられた点 (以下, 直進点) は, ユーザが通路にそってまっすぐ前へ進んでいる点である。つまり, この直進点の連続した点群 (以下, 直進パーツ) は一つの通路上を歩行している軌跡であり, 同じ線分上にある点群として判断できる。そこで図 3 のように, この直進パーツの始点と終点を結んだ線上へ同じパーツの直進点を補正する。この時, それぞれの直進点の座標は移動されるが, 第 3.1.1 章 で求めた角度差情報

は更新せず, 同じ情報を保持したままにする。これは後の第 3.1.9 章で行う進行方向の正誤チェックに用いるためである。

### 3.1.5 直進パーツの回転補正

次に直進パーツの直交化を行う。直進パーツを検出するたび, x 軸および y 軸いずれかに平行な線分へと回転補正を行う。この補正処理は図 4 のように推定軌跡の前方向から直進パーツを検出するたびに実行する。直進パーツを回転補正する際には, 補正する直進パーツ含め後ろ方向全ての推定軌跡を回転補正する。例えば, ある直進パーツが x 軸の正の方向に対して 5 度時計回りに傾いている場合, その直進パーツの始点を基準に以降全ての点に反時計回りに 5 度回転補正を加える。

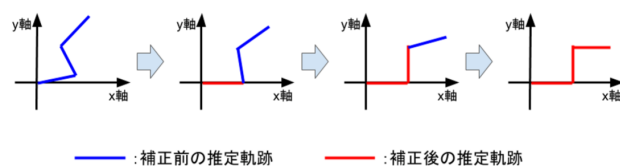


図 4 回転補正

### 3.1.6 曲進パーツの簡略化

第 3.1.2 章の分類によって「曲進」ラベルがつけられた点 (以下, 曲進点) はユーザが進んでいた通路から直交した別の通路へと移動する動作を表した推定軌跡であると考えられる。すなわち曲進点が連続した点群 (以下, 曲進パーツ) は, 直進パーツから別の直進パーツへの進行方向変化とその移動を表す推定軌跡であり, これは図 5 で表すように「2 つの直進パーツとそれに挟まれた 1 つの曲進点」として表現することが可能である。図 5 では理解しやすいよう描画を省略しているが, 直進パーツ部には複数の直進点と曲進点が含まれており, それぞれは「直進」「曲進」といったラベル情報は保持したままである。

### 3.1.7 曲進パーツの処理

曲進パーツは前後のいずれか一方に直進パーツが存在していれば簡略化を行うことが可能であり, 直進パーツは挟まれた直進パーツの延長線上に補正することが可能である。簡略化が可能な場合, 曲進パーツの中央付近の点を 1 つの曲進点とし, 曲進点の前後で曲進パーツを二分割する。二分したパーツはそれぞれ近い方の直進パーツの延長線上へと補正を行う。直進パーツの個数により処理が 2 パターンにわかれるため, 以下にそれぞれの処理を示す。

前後両方に直進パーツがある場合

前後 2 つの直進パーツそれぞれの延長線の交点の座標を求め, 曲線パーツの中央点の座標とする。その後図 6 のように, 分割した残りの曲線パーツをそれぞれ直進パーツの延長線上へと補正を行う。

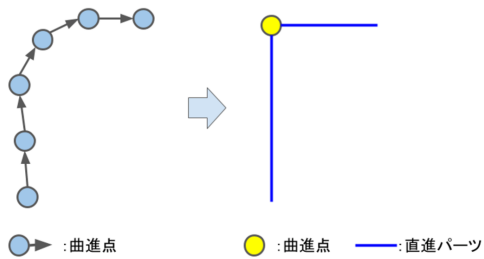


図 5 曲進パーツの簡略化のイメージ

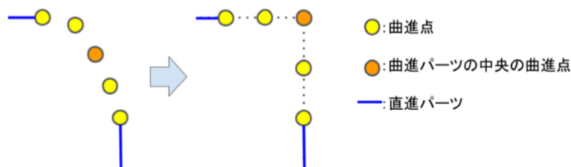


図 6 直進パーツに挟まれた曲進パーツの補正

前後いずれか一方に直進パーツがある場合

これはユーザが曲進後に意図しない歩行をした場合に起こり、「直進パーツ→曲進パーツ→ undefined パーツ」となるような推定軌跡である。直進パーツの延長線上に曲進パーツを挟んで undefined 点から垂線を下ろし、その交点を曲線パーツの中央点の座標とする。その後図 7 のように、分割した残りの曲線パーツをそれぞれ直進パーツの延長線上もしくは、垂線上へと補正を行う。

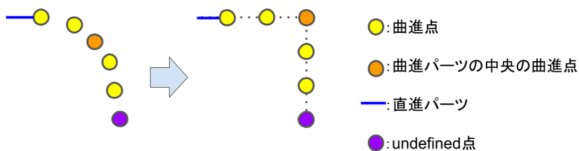


図 7 前後いずれか片方に直進パーツがある曲進パーツの補正

### 3.1.8 曲進パーツの例外

第 3.1.7 章で曲進パーツの簡略化を行う際、処理を行うべきではない例外が存在する。これは曲進パーツが非常に長い場合である。曲進パーツが長くなる要因として主に以下の 2 つの要因が考えられる。

- ・パラメータの不適
- ・U ターン

それぞれ場合について詳しく述べていく。

パラメータの不適

これは第 3.1.2 章でラベルづけに使用したパラメータがユーザに対して不適だった場合である。ユーザの歩行状態やセンサの取り付け位置等により、設定したパラメータによる座標点の分類がうまくいかず、誤ったものになってしまうことがある。例えば本来「直進」として判定されるべき点群が誤って「曲進」として判定されてしまうような場合である。これはいわば例外的な場合であり、これに対応することは本研究の本筋とは外れるため本論文では対応

しない。これに対応する方法としては、パラメータをそのユーザに対して適当なものに再設定する、もしくはラベルづけのアルゴリズム自体を全く別のものにする、といったことが考えられる。

U ターン

これはユーザが緩やかで大きなカーブを描くような歩行で U ターンをし、曲進パーツが半円のような形となった場合である。この場合、曲進パーツを挟む前後の 2 つの直進パーツが平行となるため、角度差の和から判断できる。本研究では U ターンである曲進パーツを検出した際には、undefined パーツとして処理することで U ターン部の軌跡をそのまま保持することとした。

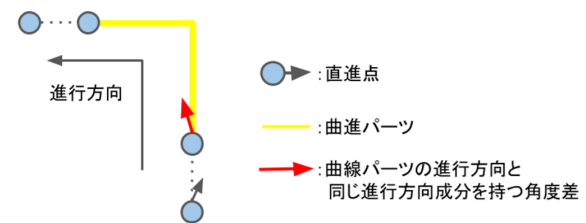


図 8 曲進パーツと同じ進行方向成分を持つ直進点

### 3.1.9 回転補正の正誤チェック

第 3.1.4 章で直進パーツは x 軸 y 軸と平行な方向へと補正を行なったが、ここで補正結果が正しいかどうかチェックを行う。これは短い直進パーツと短い曲進パーツが混在する部分や undefined パーツの存在によって、本来カーブしているはずの角度が失われてしまっている可能性を危惧したためである。まず、直進パーツのうち直進点が 5 個以上あるものを長い線分 (以下、長直線) と判断し、2 つの長直線に挟まれた区間 (以下、判定区間) を検索する。次に、検出した判定区間内の点群それぞれについて、第 3.1.1 章で求めた角度差をすべて加算し合計値を求める。ここで直進パーツの後方に進行方向変化の成分が含まれている可能性を考慮し、合計値を元に判定区間の前側にある長直線の角度差もチェックする。例えば図 8 では、曲進パーツは左方向へと進んでいるため、進行方向成分はマイナスであり、判定区間前にある直進点の角度差も同じくマイナス成分であることがわかる。この場合、赤矢印の角度差も合計値に加算する。最後に前側の長直線と後側の長直線のなす角を計算し、求めた合計値に近いかどうかチェックする。具体的には 2 つの数値の差を求め、絶対値 45 度未満であればそれらしいと判断する。差が 45 度以上あった場合、本来補正されるべき進行方向と 90 度以上違う方向へと補正されていると判断できるため、その差分後側の長直線とそれ以降すべての移動軌跡に回転補正を加える。

### 3.1.10 初期進行方向への回転補正

第 3.1.1 章から第 3.1.9 章までの一連の処理で、簡略化の処理は一通り終わっているが、推定軌跡は第 3.1.1 章で定



義した xy 座標平面基準に回転補正されているため、ユーザが測位を開始した方向 (以下、初期進行方向) が失われてしまっている。このままでは初期進行方向が x 軸 y 軸のいずれか近い方へとずれたままであるため、測位開始地点を基準に推定軌跡を初期進行方向へと回転補正を行う。ユーザの初期位置から始まる直進パーツのベクトル方向を記録しておき、簡略化が終了した推定軌跡全体をこのベクトル方向へ回転補正を行う。

## 4. 評価

本章では、第 3 章で述べた提案手法を用いた評価実験を行い、結果に対する考察を述べる。

### 4.1 評価手法

PDR の測位結果に対して提案手法を適用することで、推定軌跡や測位誤差の変化を観察した。チェックポイントを設定し、チェックポイントごとに実際の座標との測位誤差を計測し比較した。

### 4.2 実験条件

#### 4.2.1 環境

本評価では通路が碁盤目状にそれぞれが直交している環境として、兵庫県神戸市のコープ垂水の店舗を用いた。マップは図 9 に示す。図 9 上の色矢印が被験者が実際に歩行したルートであり、数字の書かれた円はチェックポイントである。0 番が測位開始地点、16 番が測位終了地点となっている。

#### 4.2.2 被験者

40 代の被験者 1 名に買い物メモを渡し、メモに書かれた商品を探してもらった。被験者が実際に歩行した正解軌跡は、被験者を後ろからビデオ撮影することで記録した。歩行軌跡は合計 364 歩であった。

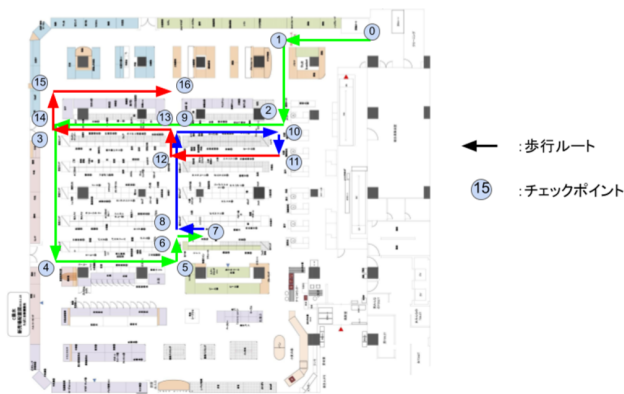


図 9 コープ垂水マップ

#### 4.2.3 センサデバイス

センサデバイスは PDR に必要な加速度、角速度が計測可能であり、一般に使用されているデバイスとして Nexus5

を用いた。センサデバイスは胸の前に固定し、測位を行った。

### 4.3 実験結果

#### 4.3.1 推定軌跡と補正後の軌跡

図 10 に PDR による推定軌跡、その軌跡を提案手法により簡略化したものを図 11 に示す。推定軌跡はそれぞれ一歩ごとに推定された座標がピンで表されている。ピンの色は第 3.1.2 章のラベルに基づいており、緑は「直進」、黄色は「曲進」、ピンクは「undefined」である。そしてオレンジは、「曲進」のうち第 3.1.7 章によって求めた「曲進部」の中点である。



図 10 PDR による推定軌跡

表 1 測位誤差

|          | 補正前         | 補正後         |
|----------|-------------|-------------|
| 平均 (m)   | 11.19456592 | 4.745506684 |
| 最小誤差 (m) | 1.878365428 | 1.563781581 |
| 最大誤差 (m) | 18.12854367 | 11.11935281 |

#### 4.3.2 測位誤差

補正前と補正後それぞれチェックポイントごとに測位誤差を図 12 に示す。また、それぞれの測位誤差平均値は表 1 に示す。なお、チェックポイント 0 番は測位開始地点のため省いている。

### 4.4 考察

実験結果に対する考察を述べる。図 10 では通路を波打ったように進む直進パーツが、図 11 では複数の線分として変換されており、ユーザの移動経路が明確なものとなった。しかし、正解軌跡と補正後の軌跡を比較すると、チェックポイント 12 番までは正解軌跡と同様な軌跡へと補正が行われているが、13 番から後の軌跡は進行方向が誤った方向へ



図 11 簡略化された推定軌跡

と補正されてしまっていることがわかる。12 番以降の軌跡を詳しく見てみると、12 番から 13 番、14 番から 15 番へと移動する部分で進行方向が正しく補正されていない。この 2 箇所の被験者の歩行を動画で確認すると、どちらにも周囲を見渡すように一回転する様子がみられ、特に 12 番から 13 番の部分では移動しながら一回転を行っていた。図 11 で対応する部分を確認すると、12 番から 13 番にはピンク色の undefined パーツが間に含まれ、14 番から 15 番には undefined パーツが見られないものの、進行方向が誤ったものに補正されてしまっている。すなわち、ラベルづけは想定通りに行われたが、第 3.1.9 章による回転補正の正誤チェックが正しく機能していないことがわかる。これは歩行しながら一回転するようなイレギュラーな歩行を想定できておらず、正確に認識し処理することができなかったためだと考えられる。次に測位誤差をみると、平均測位誤差 10.5m 以上もあったものが補正後では 4.5m 以下へと収まっており、図 12 に示されるよう、各チェックポイントでの誤差も提案手法により補正した後のほうが誤差が小さくなっている。これは実験を行なったコープ垂水のマップが碁盤目状になっていることから、提案手法の簡略化に適したマップであったためであると考えられる。さらに 12 番までの測位誤差を表 2 に示す。補正前、補正後ともに全体の平均測位誤差と比較して測位誤差は減少しているが、特に補正後の最大誤差は 11.1m から 5.1m へと大幅に減少している。補正前後の測位誤差低減率をみると軌跡全体では約 58 %、チェックポイント 12 番まででは約 62 %となっていることから、提案手法により測位誤差を大きく低減することができたと言える。

## 5. おわりに

### 5.1 まとめ

本論文では PDR による推定軌跡の簡略化を行うことで、

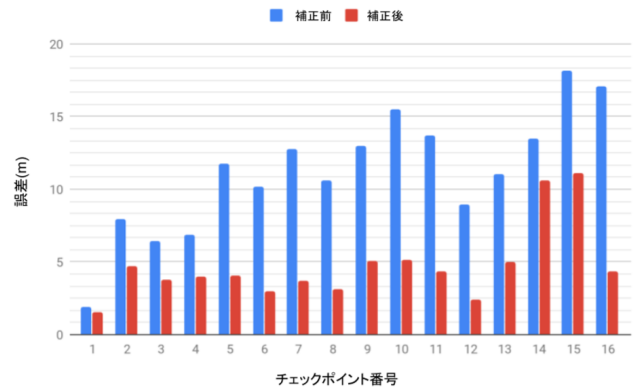


図 12 チェックポイントごとの測位誤差

表 2 測位誤差 (チェックポイント 12 番まで)

|          | 補正前         | 補正後         |
|----------|-------------|-------------|
| 平均 (m)   | 9.949320376 | 3.740415171 |
| 最小誤差 (m) | 1.878365428 | 1.563781581 |
| 最大誤差 (m) | 15.52891681 | 5.12124398  |

ユーザの移動経路理解を向上する手法を提案した。軌跡の補正には歩行状態のパターン化を行うことでパーツ分けをし、推定軌跡の線分化と直交化を行なった。実際に PDR の推定軌跡を簡略化を行うことで、移動経路の理解が容易となり、測位誤差の約 58 % の低減を確認した。

## 5.2 今後の課題

### 5.2.1 イレギュラーな歩行への対処

本手法では第 4.4 章で述べた、ユーザが一回転するようなイレギュラーな歩行には対応できていなかった。特に進行方向補正が誤ったものとなったことから、進行方向成分の抽出方法が最適でない可能性があるため、最適な手法を検討する必要がある。

### 5.2.2 undefined パーツの簡略化

本研究では undefined パーツを定義し、このパーツは簡略化を行っていない。この undefined パーツの簡略化を行うことができれば、推定軌跡をさらに簡素な軌跡へと補正ができる。undefined パーツはいわゆるユーザの予期せぬ歩行状態であるため、ユーザの歩行状態をより正確に判定できるアルゴリズムを開発する必要があると考える。

### 5.2.3 最適なパラメータ設定

本手法では一意なパラメータを設定することで、座標点の分類を行なった。しかし、ユーザの歩行特徴や歩行状態によってはこのパラメータが不適な場合がある。そこでユーザの推定軌跡から歩行の特徴や状態を認識し、それに適したパラメータを適宜設定することでさらなる精度向上が期待できる。

### 5.2.4 直交していない通路への対応

建物内の通路は直交して存在している場合が多いが、例外もあり必ずしも全ての通路が直交しているとは限らな

い. 本プログラムでは設定した  $x$  軸  $y$  軸の方向へと回転補正を加えることで直交化を行なったが, これだけでは斜め方向へと交わる通路には対応できない. そこで回転補正に用いる軸を  $x$  軸  $y$  軸からそれぞれ 45 度傾いた軸を新たに設定することで, より多くの通路状態に対応できると考える. ただし, これは補正後の推定軌跡が複雑となる要因ともなるため, 通路状態を正しく認識し, 適宜切り替える必要が生じる.

## 参考文献

- [1] M. Alfakih, M. Keche, and H. Benoudnine. Gaussian mixture modeling for indoor positioning wifi systems. In 2015 3rd International Conference on Control, Engineering Information Technology (CEIT), pp. 1-5, May 2015.
- [2] A. Ozer and E. John. Improving the accuracy of bluetooth low energy indoor positioning system using kalman filtering. In 2016 International Conference on Computational Science and Computational Intelligence (CSCI), pp. 180-185, Dec 2016.
- [3] M. Kouroggi and T. Kurata. A method of pedestrian dead reckoning for smartphones using frequency domain analysis on patterns of acceleration and angular velocity. In 2014 IEEE/ION Position, Location and Navigation Symposium PLANS 2014, pp. 164-168, May 2014.
- [4] Kazuya Miyazaki, Reina Wada, Masahiro Mochizuki, Kazuya Murao, and Nobuhiko Nishio. Evaluation of cross-assistive approach for pdr and wi-fi positioning in the real environment. In UbiComp/ISWC Adjunct, 2015.
- [5] 田川達司, 内匠逸, 打矢隆弘. カルマンフィルタとマップマッチングを用いた歩行者経路の推定精度向上に関する研究. 第 75 回全国大会講演論文集, Vol. 2013, No. 1, pp. 207-208, mar 2013.
- [6] 吉見駿, 村尾和哉, 望月祐洋, 西尾信彦. マップマッチングを用いた PDR 軌跡補正. 研究報告ユビキタスコンピューティングシステム (UBI), Vol. 2014, No. 20, pp. 1-8, oct 2014.