

スマートフォン内蔵センサーを用いた 複数フロアデッドレコニング

五百蔵重典, 鈴木孝幸, 田中博^{†1}

我々は, 屋内向けナビゲーションのための基礎技術を研究している. 本論文では, スマートフォンに内蔵されたセンサーを用いた基礎研究として, 加速度センサーを用いた歩数検出, ジャイロセンサーを用いた曲がり検出, および気圧計を用いた在階検出の方法と計測精度について述べる. そして, これらのセンサーによって得られた計測結果とマップマッチング手法を組み合わせ, スマートフォンを用いた複数フロア移動可能なデッドレコニングを行う. 実験結果として, 大学研究室があるような建物での複数フロアのデッドレコニングを行えることを示す.

Multiple Floor Dead Reckoning using a Smartphone built-in Sensors

SHIGENORI IOROI, TAKAYUKI SUZUKI, HIROSHI TANAKA^{†1}

1. はじめに

屋外でのナビゲーションシステムは実用化し, 広く普及している. 一方, 屋内向けのナビゲーションシステムの実用化のための研究は数多くなされているが, 実用化及び広く普及するには至っていないのが現状である. 屋内向けのナビゲーションの実用化への難さは, 屋外に比べて, 屋内ではより稠密な案内が求められることがあげられる. これは, 屋外で求められるよりも高精度の測位技術および高精度の地図情報が必要であることを意味している.

我々は, 「超音波による測位」および「スマートフォン内蔵センサーによるデッドレコニング」を組み合わせたプラットフォームを作成中である. すなわち高精度が求められる箇所には超音波センサーを設置し測位を行い, 精度が求められないところではスマートフォン内蔵センサーによるデッドレコニングを行うことで, 広域な測位を現実的なコストで実現することを目指している. 現在, 超音波による測位とスマートフォン内蔵によるデッドレコニングをどのように組み合わせるのかは検討中であり, それぞれの測位技術としての基礎技術を固めている.

2. 屋内測位の現状

屋内でのナビゲーションは, 数 m 間隔で並んだ集合住宅や店などの案内や, 数十 cm 間隔で並んだ商品棚への案内などへの適用が考えられるため, 屋外に比べて高精度の測位が求められている. 加えて, 屋内での測位が可能になると, 屋内での人の動線を知ることができる. そして, 人の行動の 70% は屋内であると言われているため, 様々なビジ

ネスチャンスが期待されている. これらの理由から, 多くの企業や研究者が, 屋内測位に関する様々な方法を提案している.

2.1 現在の位置情報を配信する測位方法

測位方法として, 現在の位置情報を設備側が配信し, クライアント端末が受信する方法がある. この代表的な例として GPS リピーターおよび IMES¹⁴⁾ (Indoor MESSaging System)がある.

GPS リピーターを使った方法は, 現在地を求めるのに必要な情報を (再) 配信する方法である. 具体的には, 建物の屋根に GPS アンテナを設置し, その電波を屋内に引き込み, 再放射することで実現する. クライアント端末として携帯電話を使う. そして現在普及しているほとんどの携帯電話は, GPS が内蔵されているため, 新たにクライアント端末を用意する必要がなく, クライアント側のコストが安いという長所がある. 加えて, GPS 方式は屋外でも利用できる方式であるため, 屋内外をシームレスに案内することができるという利点がある. しかし, 屋外に GPS アンテナを設置するコストは, 高い.

IMES¹⁴⁾は GPS 信号と同じ電波を使い, 現在の位置を配信する方式である. 同じ周波数帯を使っているため, ハードウェアとしては GPS 信号の受信機を利用できるという利点がある. 加えて, 屋外でも利用可能な GPS を, 屋内でもシームレスに使うことができることも利点である. しかし, 通常の GPS 信号では, 衛星の位置情報を取得し, 計算によってクライアントの位置を測位するのに対して, IMES では直接位置情報が配信されるという情報の配信方式が異なるため, クライアント端末にソフトウェア的な改修は必要になる.

^{†1} 神奈川工科大学 情報学部 情報工学科
Kanagawa Institute of Technology

現在の位置情報を配信する代わりに、現在地を識別できるように ID を配信する方法がある。この方法では、受信して得た ID から、データベースを参照し、現在地を取得する。ID の配信装置としては、アクティブ RFID タグを用いた方法がある⁵⁾。

ID を発信する機器として、電波ではなく、可視光光線を用いた実装もある²¹⁾。

これらの現在地または現在地に代わる ID を送信する測位方法の測位精度は、電波が照射されている範囲となる。つまり、電波が照射されている範囲では、同じ測位結果になる。そのため、測位精度を上げるには、電波の照射範囲を絞り、電波を発信する機器を稠密に配置する必要がある。そのため、求める精度に応じて、設置コストが高くなるという課題がある。

位置情報だけでなく、地図情報を受け取るナビゲーションシステムを著者らは開発している^{6), 26)}。通常、地図情報は膨大であるが、著者らは、各目的地への数十バイトの進行方向情報のみを送信することで、QR コードで情報を送信可能なナビゲーションシステムを実現している。情報の配信に QR コードを使うため、クライアント端末に地図情報を保持することなく、なおかつサーバーとの通信も不要、すなわちクライアント端末および各所に貼られた QR コードで動作可能なナビゲーションシステムとなっている。

2.2 距離を用いた測位

GPS のように、正確な位置が分かっている基準点と現在の位置からの距離を求めて、三角測量的に現在の位置の座標を求める方法³⁾がある。この方法は距離を求める「測距」の精度が重要になる。

2.2.1 電波の到達時間を用いた測位

電波の到達時間を用いた測距では、電波を発信する機器の場所は与えられているとする。そして、その電波発信機器から電波を受信する到達時間により距離を求める。この電波発信機器の位置と電波の到達時間の組を、3 個以上の電波発信機から得て、三角測量により、電波受信位置を求める方法である^{19), 20)}。

2.1 節で述べた方法は、電波の照射されている範囲が測位精度であった。一方、電波の到達時間を用いる測位は、理論的には正確な位置を測定可能である。しかし、電波の速度は非常に速いため、1 ミリ秒でも 299,792m 進んでしまうため、ほんの僅かな計測誤差ですら、大きな誤差を生んでしまう。そのため、10m 程度の測位精度になっているのが現状である。

2.2.2 音波の到達時間を用いた測位

電波の代わりに超音波を使って距離を求め、測位する方

式がある^{9), 13)}。(超)音波は電波よりも伝搬速度が遅いため、到達時間が長くなる。つまり計測にかかる時間が長い。高精度が期待できる方式である。しかし、音波は電波よりも到達距離が短いという欠点がある。このプラットフォームは、天井に超音波受信装置を配置し、測位する物体が超音波送信装置を持ち、天井に向けて超音波を発信する。この方式は、受信機と送信機が時刻的同期をとっていないため、音波の到達時間も未知数になる。そこで 4 個以上の受信機で音波を受信し、受信機と送信機の時刻のずれも計算によって求める。これは、GPS の受信側と送信側が逆になっている方式で、計算方法は GPS の計算方法と同じになる。この超音波を用いた逆 GPS 方式では、水平方向の精度は数 cm 程度である。

2.2.3 電波強度を用いた測位

電波強度を用いた測位とは、電波の到達時間ではなく、電波強度を用いて、電波発信機との距離を測る方法である。電波発信機器として、無線 LAN のアクセスポイントを流用することで、既存の設備が利用でき、設置コストが安くなるという利点がある。電波強度を用いて、複数のアクセスポイントとの距離が分かれば、測位が可能になる^{4), 18)}。

しかし、無線 LAN は通信のために配置するため、通常はそれほど稠密に配置しない。そこで、位置特定専用インフラ無線 LAN (Place Sticker) を用意する方法が提案されている。この方法での、測距では平均誤差 58cm での測距が可能である¹⁵⁾。

2.2.4 複数の電波強度を組み合わせた方式

位置が分かっている場所で、Mac アドレスと電波強度の組を測定する。通常、取得できる電波（無線 LAN）は、一つではないため、Mac アドレスと電波強度の組の集合を得ることができる。この Mac アドレスと電波強度の組の集合を finger print と呼ぶこととする。そして、この finger print を複数の箇所から取得し、測定した位置情報と共にデータベースに保存しておく。測位したい場合、クライアント端末は finger print を測定する。そして、この finger print とデータベース内のデータと照らし合わせ、位置を推定する方法が提案されている。この方法には Place Engine¹⁶⁾および Locky¹⁷⁾と呼ばれる方式がある。測位精度は、10m 程度であるが、さらなる研究が進んでいる。

2.3 垂直方向の移動推定

気圧センサーにより高度を知ることができることは、良く知られている。

医療機器などを製造販売するニプロは、気圧センサーを搭載することで上下の移動を判別し記録できる歩数計ウェルサポート²⁴⁾を開発し、販売している。この歩数計は、気圧センサーにより高度の変化、および移動に要した時間を

取得し、階段を上り下りしたのか、エスカレーターやエレベーターを使って移動したのかを識別する。外形寸法は48mm×78mm×18mm、質量は78g（単4乾電池2本含む）である。2008年5月初旬に発売され、価格は3万5000円である。1秒単位で、「階段上り」「走行」「階段下り」「平地歩行」「エレベーター等 Up」「エレベーター等 Down」「静止」「睡眠等（臥位）」「その他」の9種類の行動を判別して記録することで、消費エネルギーを正確に計測することを目指している。

2.4 慣性航法（デッドレコニング）

加速度センサーおよびジャイロセンサーなどの自蔵センサーを使って、慣性航法を行う研究も多く行われている。これは、屋内では、衛星からの電波が受信できないためGPSが使えないという問題、および鉄筋の建物では地磁気センサーが使えないという問題を解決するためである。また、最近では、スマートフォンに、加速度センサーおよびジャイロセンサーなどのセンサーが内蔵され、安価でこれらのセンサーを利用することができることにもよる。

屋内では地磁気センサーの値に信用性が無いため、方位が分からない。そこで、加速度・地磁気センサーを用いて、方位推定精度を上げる研究²⁾がある。既存手法が24%だったのに対して、角速度データを補正せず積分する手法を用いて81%の方位推定精度の向上が得た研究などが行われている。

加速度センサー、ジャイロセンサー、地磁気センサーなどの自蔵センサーを用いて、歩行者の位置・方位を推定する研究⁹⁾がある。この研究では、RFIDタグおよびGPSレシーバーも用いており、設置されたRFIDリーダーで、RFIDタグを読み込むことで、位置の補正も行っている。

3. 我々が行ってきた基礎研究

我々は屋内向けナビゲーションを実現するために、屋内測位に関する基礎研究を固めている。本論文では、その中で、我々が行っているスマートフォンのみを使った屋内測位の基礎研究について述べる。

はじめにスマートフォンのみで(相対的)屋内測位、つまりデッドレコニングを行うことの研究の意義について述べる。現在スマートフォンは広く普及しており、スマートフォンはその特徴から、常に携帯している。そして、近年このスマートフォンには、加速度センサー、ジャイロセンサーおよび気圧センサーなど多くのセンサーが内蔵されるようになっている。そこで、センサーを内蔵し、携帯している可能性も高く、広く普及しているスマートフォンで、歩行者向けの測位を行える可能性が非常に高い。必要に応じて、超音波などを用いた他の測位システム^{9),13)}と組み合わせる必要はあると思われるが、スマートフォンで測位でき

る範囲を広げていき、その限界を知ることは、他の測位システムと組み合わせる際の有効なデータになりうると思われる。

上記の理由から我々は、スマートフォンに内蔵された加速度およびジャイロセンサーを用い、歩数および曲がりを検出し、人の行動軌跡を描く研究を行っている。またあらかじめ用意した地図情報を元に、曲がり検出の際に現在位置を補正するマップマッチングを行っている。さらに、気圧センサーを使って在階推定も行っている。これらの研究は、他の研究者が、専用センサーで行っていたり、スマートフォンを使い単体の測位を行っている例はある⁷⁾。我々は同様の研究をスマートフォン上で、複数の方法を組み合わせた形で実現しているところに特徴があると考えている。

3.1 スマートフォン内蔵加速度センサーによる歩数推定

スマートフォン内蔵加速度センサーを使って、歩数検出を行うが、この検出方法は専用の加速度センサーを使った先行研究が元になっている。そのため専用の加速度センサーを使った研究について先に述べる¹¹⁾。歩数検出方式は、加速度センサーを用いた方法^{10),11)}を参考にしている。この方法では、まず加速度の値を3軸合成する。3軸合成した値をCとする。サンプリング間隔は、研究当時のスマートフォンの性能で十分測定可能な60msで設定した。直近3回の3軸合成値を使うため、最新値を C_n とし、新しい順に C_{n-1} および C_{n-2} とする。そして、この3軸合成した値を加重平均した値 $S = (3C_n + 2C_{n-1} + C_{n-2}) / 6$ を、歩数検出に用いる。検出アルゴリズムは以下の通りである。

1. Sが閾値 α (手・胸ポケット:1050mG, 腰:1100mG)より高くなることを検出。
2. 1秒以内にSが閾値 β (手・胸ポケット:950mG, 腰:1000mG)より低くなることを検出した場合に歩数としてカウントアップ。
3. 上記が1秒以内に検出、または上記が検出できず1秒経過した場合は1.に戻って繰り返す。

実験に使用した専用センサーは以下の通りである。

表1 専用センサーの仕様

型名	ワイヤレステクノロジー社／WAA-006
通信機能	Bluetooth Ver2.0+EDR
通信距離	約10m
寸法／重さ	39×44×12(mm)／20(g)
稼働時間	約6時間
加速度センサ	3軸加速度センサ (取得範囲／分解能)：±4G／2mG
ジャイロセンサ	3軸ジャイロセンサ (取得範囲／分解能)：X,Y軸(±500deg/sec), Z軸(±300deg/sec)／X,Y軸(1.46deg/sec), Z軸(4.37deg/sec)
サンプリング周期	最大333Hz

上記のアルゴリズムで、歩数検出の精度実験を行った。

取り付け位置は、様々な位置でスマートフォンを保持する可能性を考慮し、手持ち、胸ポケットに入れた場合、腰に装着した場合で実験を行う。手持ちに関しては、地面に対して水平（-10～10 度）に持つことを想定したパターン A および、画面を顔に向けながら持つことを想定したパターン B（地面に対して 30～50 度）で行う。実験は、一直線に 100 歩歩行した時の、前記の表 1 に示すセンサーを用いて、検出歩数を数える。計測被験者は 5 名、それぞれ 10 回ずつ取得を行う。10 回分の平均と標準偏差を表 2 に示す。結果から、慣性センサーの保持／装着状況に関わらず、およそ 99% の精度で歩数検出が行えることを確認した。

表 2 専用加速度センサーによる歩数検出精度

被験者	センサ状態	手		腰	胸ポケット
		パターン A	パターン B		
A	平均	99.6	99.4	99.0	99.4
	標準偏差	0.5	1.0	1.1	0.5
B	平均	99.5	99.7	98.7	99.4
	標準偏差	0.5	0.5	1.2	0.5
C	平均	99.6	99.4	99.5	98.8
	標準偏差	0.5	0.7	0.5	0.9
D	平均	99.8	99.8	99.5	99.4
	標準偏差	0.4	0.4	0.7	0.5
E	平均	99.3	99.4	98.8	99.5
	標準偏差	0.7	0.5	1.0	0.5

単位: 歩

上記の研究結果を踏まえて、 GALAXY NEXSUS (SC-04D) SAMSUNG 製、OS: Android 4.0.3 で同様の実験を行った。実験方法は、専用センサーを使った方法と同じであるが、装着位置は手のみの 2 パターンとなっている。

表 3 スマートフォンによる歩数検出精度

被験者		センサ状態	
		A パターン	B パターン
A	平均	99.8	99.1
	標準偏差	0.8	1.2
B	平均	99.8	99.6
	標準偏差	1.3	0.7
C	平均	97.9	100.7
	標準偏差	2.2	3.3
D	平均	99.9	99.8
	標準偏差	1.2	0.4
E	平均	95	85.4
	標準偏差	2.2	4.4
平均		98.5	96.9

専用センサーを使った方式に比べて、やや精度が落ちるものの、スマートフォン内蔵センサーにより、歩数検出が可能なが分かる。

スマートフォン内蔵センサーを使った歩数の検出精度が少し落ちるのは、スマートフォンのセンサーの精度に寄るのではないと思われる。例えば、スマートフォンを用いた加速度センサーの値の取得の場合、サンプリング間隔が一定にならないという問題がある。GALAXY NEXSUS (SC-04D) SAMSUNG 製で、加速度センサーの値を取得した

ところ、Android のバージョンで以下のような差異がある。サンプリング間隔は、低速を選択している。ただし、Android 端末はそれぞれ別の物を使用しているので、個体差による差もあると思われる。表 4 に、100 秒間加速度センサーの値を取得した場合の、Android 端末のバージョンによるサンプリング間隔の差異を示す。

表 4 Android 端末のバージョンによるサンプリング間隔の差異

Ver.	4.2	4.1.1	4.0.3
平均	0.064675871	0.064499016	0.072349493
分散	0.001107155	0.001086558	0.010655972

(単位: 秒)

サンプリング間隔の分散を見ると、4.0.3 は、4.2 および 4.1.1 に比べて約 10 倍の広がりを持っている。当時の実験は、4.0.3 で行っていたが、4.2 で行くと、また違った結果が出てくる可能性があると思われる。

3.2 ジャイロセンサーによる曲がり推定

我々はスマートフォン内蔵のジャイロセンサーを用いた、曲がり判定に関する研究を行っている。

3.2.1 角度による曲がり判定

ジャイロセンサーから得られる角速度を時間積分して曲がり角度を求めることで、曲がり判定を行う。角速度を時間積分することで、ローパスフィルターを施したのと同じ効果を期待している。

ジャイロセンサーの床面と鉛直方向になる軸、つまり手に保持した場合は Z 軸、腰、胸ポケットに装着した場合は Y 軸の出力を用いる。具体例として、左曲り、曲率半径 1m、手で地面に対して水平にスマートフォンを保持した場合の角速度波形を図 1 に示す。

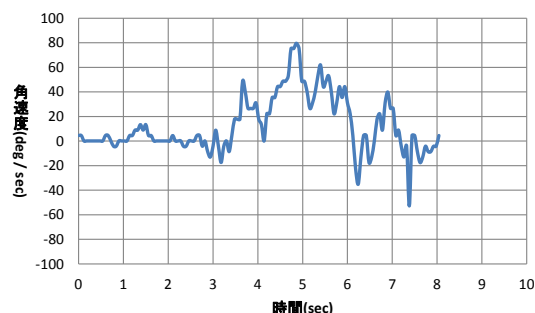


図 1 角速度の波形

図 1 の角速度の波形を時間積分すると、図 2 に示す図となる。

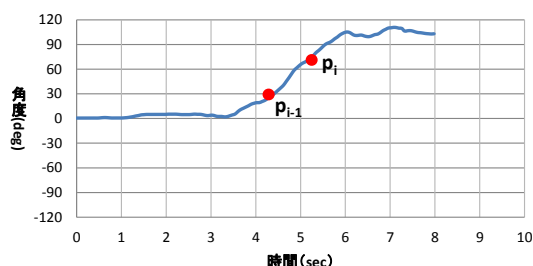


図 2 角度の波形

経過時間 i 秒のときの方向角度を P_i とする．曲がり判定は、 $|P_i - P_{i-1}| > 15$ の場合に曲がり検出を開始する． $P_i - P_{i-1} > 15$ の場合は左曲がり、 $P_i - P_{i-1} < 15$ の場合は右曲がりとする．曲がり検出中に $|P_i - P_{i-1}| < 15$ となった場合に、曲がり検出判定を終了する．言い換えると、1 秒間に 15 度以上の曲がりを検知した時に、曲がったと判定する．

ジャイロセンサーを手に保持し、曲がり判定を行う実験を行う．実験に使用したセンサーは、歩数検知に用いたセンサーは表 1 に示すものを使用する．コーナリング角度 90° を歩行する際には曲率半径毎に貼ったテープ上を歩行する．今回定めた曲率半径は 0.5m, 1.0m, 1.5m, 2.0m, 2.5m の 5 種類である．これは、個人ごとによって曲がる時の曲率半径に差があることを考慮したためである．被験者、5 人に対して、右曲がり、左曲がりを、各 10 回の実験を行う．実験を行った結果 100% の曲がり判定結果を得ることができた．角度を 45° の曲がりとし、同様の実験を行った場合、98.5% の曲がり判定精度を得ることができた．また、ジャイロセンサーを胸に保持した、 90° の曲がりを実験を行った場合、曲がり判定は 99% の精度となった．腰に装着した場合は、検出ができなかったため、実験は行わなかった．

専用センサーを用いた同様の処理をスマートフォン内蔵センサーにも実装し、実験を行う．実験は、スマートフォンを手に持ち、 90° の曲がり判定を行う実験を行う．被験者は 5 人で、右回りおよび左曲がりを各 10 回行う．その結果、曲がり判定結果が 100% になることが分かった．

角度による曲がり判定では、スマートフォンを手に持った場合においては曲がりを正しく検出できるものの、腰などの他の部位に装着した場合には曲がりを正しく検出できないという問題を発見した¹¹⁾．これは、歩行時に腰がひねる動作をすることによって曲がり判定に悪影響を及ぼしていることが原因だと考えられる．



図 3 歩行時の腰の動き

我々は、スマートフォンを胸や腰に固定している場合に

比べて、スマートフォンを手に持っている場合、手の振動などが悪影響を及ぼすことを懸念していた．しかし、曲がり検出（ジャイロセンサー）に関しては、腕がダンパーの役目をし、様々なノイズを打ち消すようである．

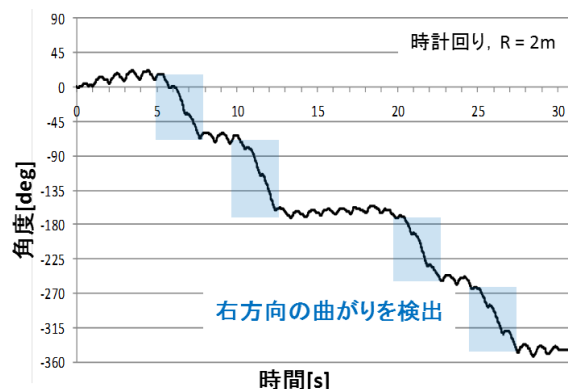


図 4 スマートフォンを手に持った時のジャイロセンサー値

3.2.2 角速度による曲がり判定

本稿では、腰のひねり運動の影響を受けずに曲がりを判定できるアルゴリズムを提案するとともに、曲がり検出精度を実験によって確認した結果について述べる

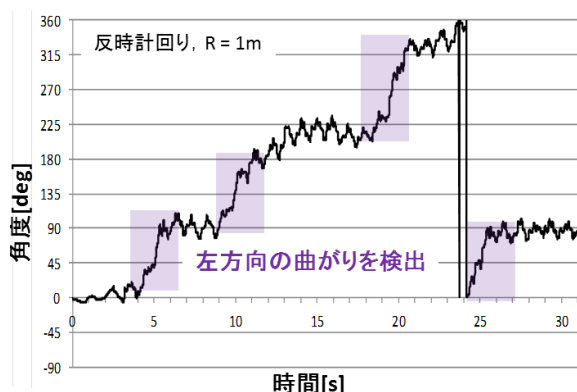


図 5 スマートフォンを腰に装着した時のジャイロセンサー値

スマートフォンから得られる角速度に 2 次の IIR フィルタ²²⁾を適用する．IIR フィルターの時定数および Q 値は、実験データから、それぞれ、時定数は 0.01 とし、 Q 値を 0.5 と設定する．

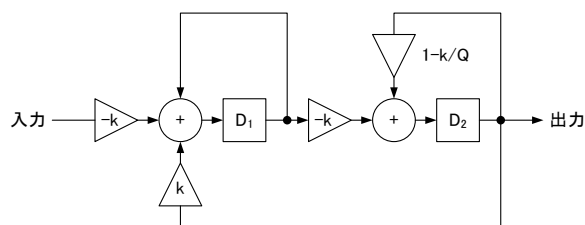


図 6 状態変数型 2 次 IIR ローパスフィルター

この IIR フィルターによって腰のひねる動作などによる影響を除去する．

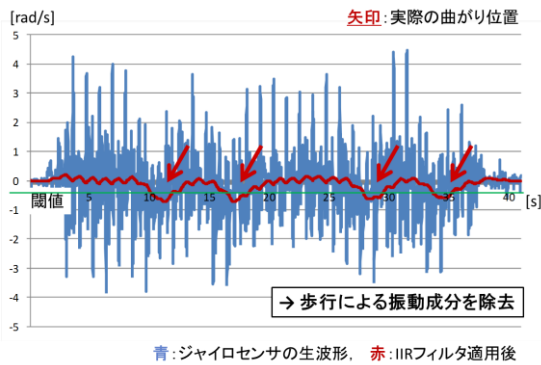


図 7 ジャイロセンサの生波形と
IIR フィルター適用後の波形

そして IIR フィルター適用後の角速度が閾値 $0.3[\text{rad/s}]$ を超えた場合に曲がりの可能性がある（曲がり候補）と推定する。曲がり候補の状態が 1 秒以上継続した場合に曲がり開始と判定し、この状態でさらに角速度が閾値 $0.3[\text{rad/s}]$ を下回った場合に曲がり終了と判定する。

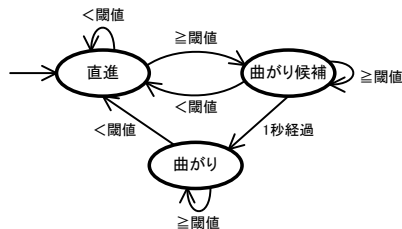


図 8 角速度による曲がり候補の状態遷移

被験者 5 名に対して、スマートフォンを決められた位置（手、腰、胸、腕）に所持・装着し、一定の経路上を歩行させ、曲がり検出回数を計測した。歩行経路には 90° の曲がり角が 4 か所存在し、それぞれの所持方法で時計回り（曲率半径 1m, 2m）及び反時計回り（同 1m, 2m）を各 10 回、合計 40 回歩行することとした。

角度による曲がり判定では手持ちの場合では検出精度が高いが、他の部位では精度が低下する。特に腰では逆方向の曲がりの誤検出が見られることもあった。一方、角速度による曲がり判定では所持位置によらず、逆方向の曲がりを検出するなどのエラーも発生せず、100%の検出精度を得ることができた。

3.3 マップマッチング

地図情報があると、測位情報を様々に補正することができる。これをマップマッチングと言う。カーナビゲーションシステムは、GPS による測位と地図情報を組み合わせたシステムである。

我々は、マップマッチングをするために、角（コーナー）の座標を地図情報に含めている。そして、相対的自己推定をするために、現在のスタート位置および進行方向を保持している。そして 3.1 節で述べた歩数判定により、歩行を

検知し、1 歩あたり 85cm 進むように現在位置を更新する。1 歩 85cm に設定した根拠は、専用センサーを使った先行研究¹¹⁾を元になっている。この研究では、被験者 5 人で 100m の直線を歩いた時の歩数を元に、平均を求めて 1 歩あたり 85cm が妥当であることを検証し、デッドレコニングを行っている。3.2.2 項で述べた曲がり判定により進行方向を更新していく。そして、曲がり判定により進行方向を更新する際には、現在位置を現在地から最も近い角の座標に変更する。

実行例を示す。小さい黄色い丸は、角の情報である。赤い点が移動軌跡を表している。青い丸が、マップマッチングによって補正された座標を持つ移動軌跡を表している。

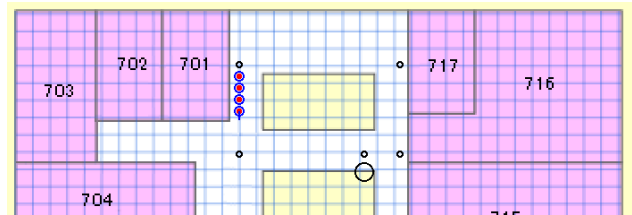


図 9 マップマッチングの実行例 1

図 10 に、歩数による距離推定が短く出てしまった例を示す。曲がり判定により、曲がりが検出されたため、現在地を補正している。赤い点は、補正されていない移動軌跡のため、壁を通り抜けたかのような軌跡が表示されている。青い丸は、マップマッチングによる補正が働き、角を正しく曲がったように軌跡が修正されて表示されている。

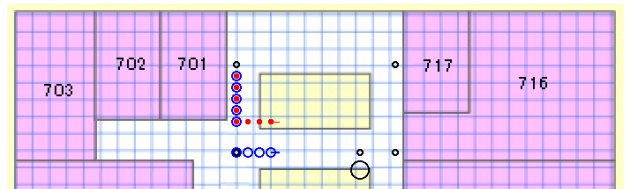


図 10 マップマッチングの実行例 2

歩数による距離推定の誤差は、歩数が多くなればなるほど大きくなる可能性が高い。そこで、歩数の多さが、マップマッチングにどのように影響を及ぼすかを調べる実験¹²⁾を行う。実験場所は、本研究室がある建物で行う。1 番長い直線距離 27m5cm 先の角まで歩いた後、曲がる場合と、2 番目に長い距離である 20m81cm 先の角まで歩いた後、曲がる場合の実験を行う。

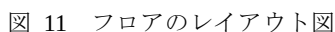


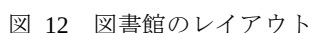
表 5 に示す.

表 5 に示す通り，曲がり判定は，正しく成功した．また

表 5 マップマッチングによる補正

被験者(歩幅)	曲がり箇所	曲がり成功回数		修正距離平均		標準偏差		歩数平均	
		右	左	右	左	右	左	右	左
A(71cm)	2081cm先	10	10	53.6	18.1	59.9	62.2	28.4	28.9
	2705cm先	10	10	-3.1	25.3	70.6	47.9	38.1	37.7
B(75cm)	2081cm先	10	10	-142.5	-67.5	53.0	81.0	29.5	28.5
	2705cm先	10	10	-238.0	-200.5	47.4	50.6	39.2	38.7
C(72cm)	2081cm先	10	10	-140.6	-25.2	96.4	104.3	30.7	29.1
	2705cm先	10	10	-127.6	-98.8	83.5	124.5	39.3	38.9
D (65cm)	2081cm先	10	10	94	68	69.9	79.9	30.4	30.8
	2705cm先	10	10	63	17.5	82.2	43.9	40.6	41.3
E (82cm)	2081cm先	10	10	-201.4	-184.2	55.3	66.9	27.7	27.6
	2705cm先	10	10	-323.8	-323.8	81.5	71.8	36.9	36.9

図書館のように棚が整列配置されている場所は、曲がり判定の修正候補が複数ある。誤補正しやすくなると思われる。そこで、図書館でのマップマッチング実験を行う。図書館のレイアウトは図 12 の通りである。



歩く距離が長いほど、誤差が大きくなるため、この誤差

を補正できるかを確認する実験を行う。そこで、1直線上に歩いた後、曲がり、正しくマップマッチングによる補正が行われたかを確認する実験を行う。曲がり角の情報は、1直線状に配置しており、配置間隔は0cmから168cm刻みで13か所である。実際に曲がった箇所は、曲がり角の6番目から10番目に該当する。すなわちそれぞれ1008cm, 1176cm, 1344cm, 1512cm および 1680cm である。実験結果を表6に示す。

表 6 図書館での曲がり判定の実験結果

	歩数平均		修正距離平均		曲がり成功回数	
曲がり箇所	右	左	右	左	右	左
1008cm	13.2	13.3	-24.6	-32.1	3	3
1176cm	16.1	15.9	-13.3	1.3	6	5
1344cm	17.6	17.7	6.7	-0.9	4	4
1512cm	19.7	19.8	-37.3	-45.7	1	1
1680cm	21.3	21.8	-13.0	-27.7	0	1

単位: 歩 単位: cm (n/10)

図書館のように曲がり判定の箇所が複数存在する場合は、正しくマップマッチングできていないことが分かった。これは、人間はある程度の曲率で曲がるためと思われる。図書館のように曲がり角が細かく設定されている場合、図 12 の上部のオレンジ色の丸で示すように、曲がりを検知する箇所が、本来の曲がりよりも手前の位置になる。そのため、手前の曲がり角を補正值として認識してしまいという、間違った補正が行われてしまう。これは曲がり判定の開始を曲がり開始位置にしていることが考えられる。曲がり判定と判断した箇所を、曲がり判定区間の真ん中にする事で、適切に補正することができると考えられる。

3.4 気圧センサーによる階数推定

大気圧は高度が上昇すればするほど、ほぼ線形に下がっていくことは良く知られている。この事実を元に、気圧計を元に相対高度を求める高度計は広く普及している。理科年表¹⁾によると、気圧と標高と気温には、次のような関係式²⁵⁾が成立し、気圧から標高を求めることができる。

$$h = \frac{\left(\left(\frac{P_0}{P_1} \right)^{\frac{1}{5.257}} - 1 \right) \times (T + 273.15)}{0.0065}$$

式(1)で h : 標高(m), P_0 : 海拔 0m の気圧 (1013.25hPa), P_1 : 標高 h (m)での気圧 (hPa), T : 標高 h (m)での気温($^{\circ}\text{C}$)である。

本実験は、Android 端末 Galaxy Nexus(OS:4.1.1)で行う。
この Android 端末の気圧センサーの分解能は、0.01Pa よりも細かくとることが可能である。

予備実験として、神奈川工科大学情報学部棟 B1F から 13F において、Galaxy Nexus(OS:4.1.1)の気圧センサーで計測した。1 階層当りの標高差 4.0m であり、サンプリング間隔は、Normal モードを指定する。測定時間は、各階 30 秒

で行う。その結果、下記の知見を得た。

- ・1階層[4.0m]ごとの気圧差の平均は 44.6～48.4Pa
- ・気象・時間経過による気圧変化は、大きく早い時に 3 分間で 40Pa
- ・1階層分の移動でも、気圧の変化を確認できる
- ・1階層分の移動時間は階段で意識してゆっくり登ったときでも 20 秒程度

加えて、エレベーターで移動した時の気圧の変化を見た結果以下の知見を得た。

- ・専用センサーを用いたときの直線回帰の相関係数は 0.82～0.93
- ・Android の気圧センサーを用いたときの直線回帰の相関係数は 0.69～0.88 である。

これらの知見から、気圧の変化が気象・時間経過によるものなのか、階層移動によるものなのかを判別することが可能であることが分かった。つまり、気圧の変化により、階の移動を検知することが可能である。

得られた知見をもとに閾値を 45Pa と決定し、気圧センサーの値から在階数を推定するアルゴリズムを考案し（図 13 にフローチャートを示す）、Android 端末 Galaxy Nexus(OS:4.1.1)に実装して、評価実験を行った。

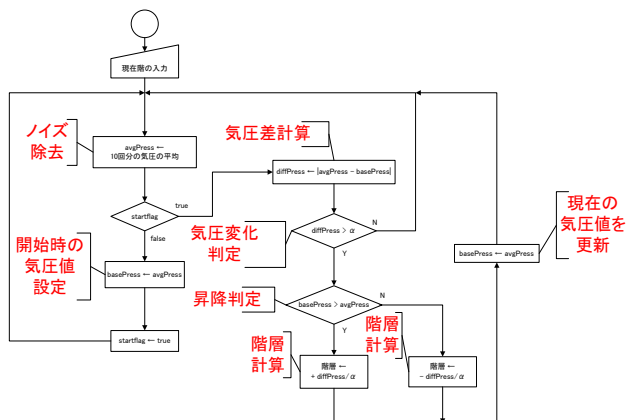


図 13 気圧センサーの昇降判定アルゴリズム

サンプリング間隔は FASTEST を指定し、約 75msec である。実験場所は、13 階建ての神奈川工科大学情報学部棟（1 階層の高さ 4.0m）で、B1F～13F の階段の昇り下り×3 セット、エレベーターの昇り下り×2 セットを行い、(1) 在階の判定成功率と、(2) 前階層から規定階層分の変化の検知率の評価を行った。実験結果を表 7 に示す。閾値の一階層毎の気圧差 α の値は、45Pa を使用する。

表 7 気圧センサーを用いた在階推定の実験結果

	(1) 平均判定成功率	(2) 階層変化検知率
階段昇り	29.6%	92.3%
階段下り	98.6%	100.0%

エレベーター昇り	36.1%	92.3%
エレベーター下り	99.2%	100.0%

気圧センサーによる在階推定システムを作成し、実験を行った結果、まだ実験回数は少ないものの 96.2%以上の確率で在階を推定できることを確認している²³⁾。誤認識の原因として、誤差の累積により、検知率が低下することを確認している（参照：図 14）。

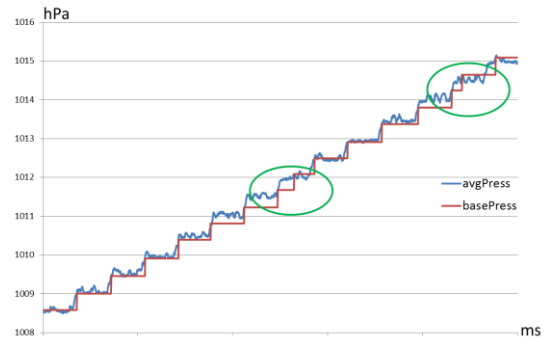


図 14 気圧センサーの値と在階推定の基準の気圧

本節をまとめると、我々はスマートフォンにも気圧センサーが内蔵されていること、およびこの気圧センサーの精度が高精度であることに目を付け、ビルの各階への移動が、気圧の変化として取得できることを突き止めた。

なお、本実験では、Android 端末 Galaxy Nexus(OS:4.1.1)を用いて実験を行った。この機種では、600Pa(6hPa)程度のスパイクノイズが起きることが分かった。在階推定時では、1階層あたりの標高差で 44.6～48.4Pa を計測しているため、この 600Pa のスパイクノイズは非常に大きく、過去数個の平均を取ることで、除去することができない。そこで前回取得した値から 100Pa(1hPa)以上の値の変化があったときは、その値を使用しないというノイズ除去機能を追加している。

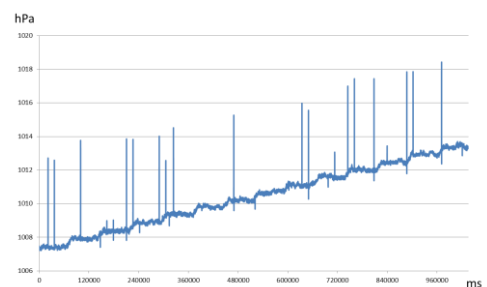


図 15 気圧センサーで検出されるスパイクノイズ

なお、このスパイクノイズは、WiFi および Bluetooth をオフにしていると、出ないことを我々は確認している。

3.5 超音波による逆 GPS 方式による測位方法

本項では、超音波による逆 GPS 方式による測位方法について述べる。この方法は、スマートフォンによるデッドレ

コニングを補うために、外部システムとの連携を図るための方法である。

この測位方法は、超音波の受信センサーを天井に設置し、測位対象が超音波を発することで、測位を行うシステムである。水平方向では、最大でも 10cm 程度の誤差という高精度の測位を実現している。原理的には GPS の方式と同じであるが、測位対象が超音波を発しているため GPS とは波の進行方向が逆である。そのため逆 GPS 方式と呼ばれている。音波は電波に比べて減衰が著しいが、音波は電波に比べて、100 万倍程度低速であるという特徴がある。速度が遅いという特徴は、到達時間を求めやすいことにつながり、到達時間から距離を求める（測距）値が正確になるという特徴がある。この特徴をいかし、高精度の測位を行っている。

しかし、この方式は測位したい箇所に稠密に超音波受信機を設置しなくてはならず、設置コストが高いという欠点がある。そこで、設置コストを削減するために、スマートフォンによるデッドレコニング技術と組み合わせることで、お互いの欠点を補いあい、広域に高精度の屋内測位が可能になると考えている。

4. 本研究内容

我々は、スマートフォン内蔵加速度センサーおよびジャイロセンサーによる単一フロアのデッドレコニングを行った¹²⁾。これに我々が開発した気圧センサーによる在階推定(3.4 節)の技術を組み合わせることで、複数フロアでのデッドレコニングを可能にすることを目指す。

4.1 マップ情報

従来は、地図情報には、角の XY 座標のみを保持していた。これに加えて、階段の情報を含める。階段の情報は、階段の位置を示す X 座標、階段の位置を示す Y 座標、階段に進入する方向、登り階段の歩数、下り階段の歩数の 5 つ組から構成される。登り下りとも「歩数」であり、「ステップ数」ではないことに注意が必要である。さらに、従来はフロア情報が 1 フロア分であったが、必要な複数フロア分用意する必要がある。

4.2 階段による移動

本研究では、フロア移動を伴うデッドレコニングを行うため、フロア内の移動と階段移動を切り替えなくてはならない。本研究では、階段移動は、階段移動とだけ判定することとし、階段移動中はどこにいるか特定しないこととする。そのため、階段を引き返すなどは無いことを仮定する。また、階段によりフロアを移動したとき、XY 座標は同一で高さが異なる位置に移動する構造を仮定している。

4.2.1 階段開始判定

階段移動を開始したかどうかは、階段に近づいたかどうかで判断する。このとき、進入方向も階段へ向かっている方向であるかも階段移動を開始したかの条件に含める。進行方向を階段移動開始の条件に含めるのは、階段移動の前も階段移動終了直後も階段のそばにおり、これらがどちらなのかを判定ができないためである。

4.2.2 階段終了判定方法の検討

階段移動の終了判定方法は、加速度センサーを使う方法、気圧センサーを使う方法、ジャイロセンサーを使う方法、地磁気センサーを使う方法などが考えられる。

加速度センサーを使う方法について述べる。加速度センサーを使うことで、歩数を検知することができる。階段のステップは、1 歩で進むことが圧倒的に多いため、歩数によって階段による移動を行ったかを判定する方法は、非常に有用に思える。ただし、階段の踊り場を何歩で歩くかは、人により差があるため、踊り場を歩く歩数で誤差が生じる可能性がある。

気圧センサーを使う方法について述べる。気圧センサーを使うことで、フロアの移動を検知することができる。これを利用して、1 フロア分移動したかどうかを判定する方法が考えられる。この方法での検証は正確には行っていないが、数ステップのずれがどうしても出てしまう。また、誤差が大きいときは、1 フロアの半分ほどのずれが生じることもある。気圧センサーはかなり高精度であるが、階段移動終了判定が出来るほどは、高精度ではない。

ジャイロセンサーを使う方法について述べる。階段で、フロアを移動したとき、同一 XY 座標に移動することを想定している。このような階段の構造の時、4 回の曲がりを経て、階段開始時と背中合わせの向きを向く。

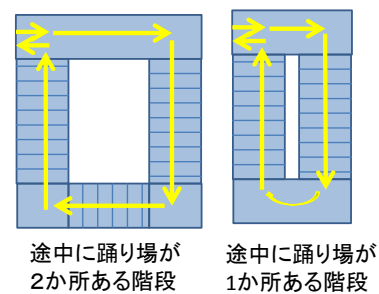


図 16 踊り場の構造

ジャイロセンサーを使い曲がり角を検知することができるため、曲がり角検知を使うことで、階段を 1 周したかどうか、すなわち階段移動終了後に進行方向が 180 度変わったかどうかを判定する方法が考えられる。しかし、我々の曲がり角検知ロジック(3.2 節)では、通路などの曲がり角を想定した検知方法であるため、90 度の曲がり角を想定している。そのため、途中に踊り場が 1 か所ある階段などのように、180 度の曲がり角を判定できない。また、曲がり角がそれほど

頻繁に起きることを想定していないため、直近 2 秒間の曲がり角度を調べたり、曲がり角が 1 秒以上継続した場合に曲がり角と判定するなどしており、短時間で急激な曲がり角を検知できないという問題点がある。

地磁気センサーを使う方法について述べる。ジャイロセンサーを使った方法と同様に、階段を 1 周したかどうか、すなわち階段を終えたときに進行方向が 180 度変わったかどうかを、地磁気センサーを使って判定する方法が考えられる。しかし、本実験環境では、地磁気センサーが適切に動かなかった。これは、実験環境が鉄筋のビルであるため、鉄筋の影響で、地磁気センサーに狂いが生じるためと思われる。

様々な方法を検討したが、加速度による歩数検知により、階段を移動するのに必要な歩数を数えることで、階段の終了判定をする方法を採用する。

4.2.3 階段の終了判定の実装方法

階段での移動の終了判定では、歩数検知と在階推定が必要である。歩数検知には加速度センサーの値が必要であり、在階推定には気圧センサーの値が必要になる。値の変化はイベントの形で取得可能である。2 種類の値をイベントの形で取得しなくてはならないため、以下のように実装した。まず、階段での移動が開始したと判定した場合、在階推定の処理ルーチンを起動する。このルーチンは、起動したときの気圧値を覚えておく。そしてこの覚えておいた気圧値と、現在の気圧値との差から、階の移動状況を推定する。推定した結果は、変数に入れて参照可能な形にすることで、他の処理中でも、在階推定を使えるようにする。

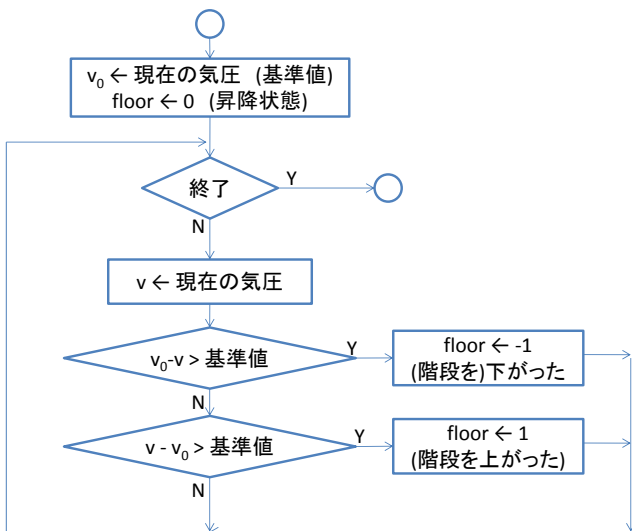


図 17 在階推定の処理

階段の終了判定は、階段を歩いた歩数で行う。階段によって登りきる（降りきる）歩数は異なるため、地図情報には、階段ごとに登りきる歩数（降りきる）を登録している。加えて、同じ階段でも登りと降りでは、歩数が違うことが

予備実験で分かっているため、登りと降りの両方の歩数を格納しておく。階段を終了したときは、マップマッチングとして補正をかける。具体的には、現在地を階段のある位置にし、進行方向を階段に向かう方向とは 180 度逆向きに設定する。

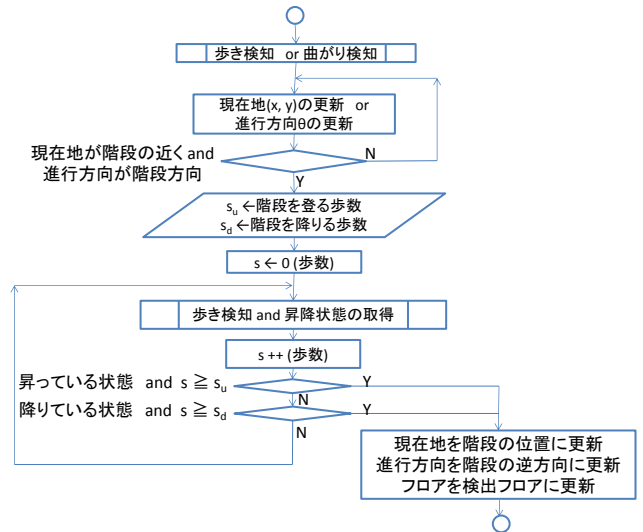


図 18 階段への移行処理、階段の終了処理の流れ

5. 実験

本論文での実験は、Samsung 製 GALAXY NEXUS(NTT docomo 発売 SC-04D) で行う。OS のバージョンは Android 4.2.2 である。

5.1 階段の上り下りによる歩数の違い

階段を降りるときは、重力による加速がつきやすいため、加速に身体を委ねると、身体をコントロールするのが難しくなる。そのため、移動速度を抑え気味にし、かつ身体のバランスを出来るだけまっすぐに保とうとする。そのため、踊り場のような箇所で、足をそろえるような動きをする。つまり、階段の進行方向に対して、身体を正対させる動きをする。この階段の進行方向に身体を正対させる行為は、踊り場に右足から降りたときも、右足から降りたときも変わらず起こる。

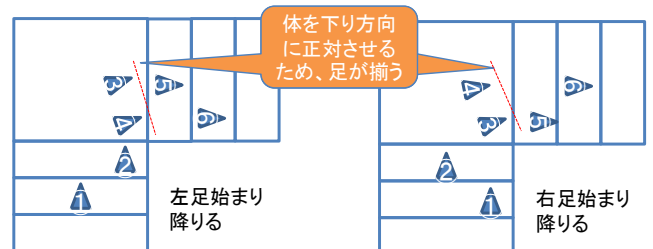


図 19 階段を降りるときの足の運び方

階段を登るときは、重力に逆らって力を使って登る。そのため、出来るだけ省力で登ろうとする。そのため、踊り

場での足を置く回数を1歩にとどめようとする。図にあるように右回りの踊り場に、左足で降りたときには（図 20 の右側）、踊り場に足を大きく踏み出し、踊り場には1歩で歩けるようにする。右回りの踊り場に、右足で降りたときにも、踊り場は1歩で歩く。

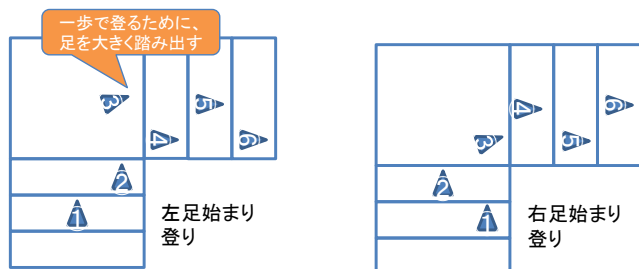


図 20 階段を降りるときの足の運び方

上記の理由から、階段を降りるときは、登るときに比べて歩数が多い傾向がある。

5.2 階段での歩数検知

我々は平地での歩数検知のみを行ってきたため、我々が開発した方法で階段の歩みを正確に検知できるか分からない。そこで、階段の歩みをきちんと測定できるかの実験を行う。実験した階段は、本研究室が存在するためものにある北階段および南階段である。階段の構造は、図 16 に示す2種類の構造をしており、踊り場が1か所のタイプが北階段、踊り場が2か所のタイプが南階段である。5人の被験者に各5回、階段の登りおよび階段の下りを歩いてもらう。実際の歩数およびスマートフォンによる歩数検知を比較する。

表 8 歩数検知

	昇降別		a)アプリ	b)実歩数	差(a-b)
北階段	登り	平均	29.32	29.20	0.12
		分散	1.34	0.64	0.35
	下り	平均	29.40	29.24	0.16
		分散	1.92	1.14	0.77
南階段	登り	平均	28.60	28.40	0.20
		分散	4.48	1.20	2.48
	下り	平均	29.44	29.32	0.12
		分散	2.01	1.26	0.35

スマートフォンを用いた歩数検知と実歩数の差は、平均で0.1歩程度の差しかない。その為、階段での歩数検知は十分機能していると言える。しかし、南階段で登りの分散値が4.48と非常に精度が悪い。データを精査すると、一人の被験者で1回だけ、スマートフォンによる計測が35歩、実歩数が29歩となっており、スマートフォンによる計測が大幅にずれているデータがあった。これが影響しているが、このデータ以外は、ほぼ正しく計測できている。そのため、実験に用いた結果が少ないため、このデータが影響しているものと思われる。

また、5.1 節で述べた登りと下りでの階段の差があることも実験によって確認することができた。

5.3 マップマッチングに必要な歩数

階段による移動を終了した時に、マップマッチングによる補正を行う。そのため、階段による移動を終了したことを正しく判定することは、移動後のフロアでのマップマッチングが正しく行うために非常に重要である。

階段移動の終了判定には、階段での移動歩数を使うと述べた。階段による移動歩数は、表 8 の実験結果を見ると、歩数の誤差は、ほぼ1,2歩に収まっていると言える。そのため、この1,2歩の精度で階段の移動の終了が検知可能であると言える。

階段での移動は、登りと下りで周る方向が異なる。この周る方向の違いで、最後の平地で歩く歩数が異なる。

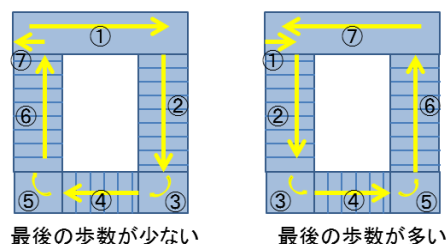


図 21 階段の登り下りでの、最後の踊り場の歩数の違い

階段移動終了後のマップマッチングでは、位置だけでなく移動方向も補正される。つまり正しく補正されるためには、図 21 の⑦に示す箇所、既定の歩数に達していないといけない。そこで、階段移動に必要な歩数だけでなく、踊り場で何歩歩いたかも、5.2 節の実験の時に、計測している。実験結果を表 9 に示す。

表 9 最後の平地を歩いた歩数

	昇降別		最後の踊り場
北階段	登り	平均	2.68
		分散	0.22
	下り	平均	3.96
		分散	0.04
南階段	登り	平均	3.04
		分散	0.20
	下り	平均	2.56
		分散	0.33

歩数の平均は、2歩から4歩程度である。階段の空間に入ってきた瞬間から数え始め、階段の空間を出た瞬間で数え終えている。そのため、実際はこれよりも1,2歩追加した値の範囲で、歩数による階段の終了を検知することができれば良い。

2名で各5回、北階段および南階段の登りと下りを正しく検出できるかを確認したところ、正しく認識できることを確認している。

6. 考察と今後の課題

実験により、複数フロアの移動を検知し、マップマッチングを行うことができることが示された。その他にも実運

用に向けては、さらに検討すべきことがある。

6.1 階段の上り下り

階段の昇降時の歩数は、従来の歩数検知を使っている。従来方式は平地を対象にしている。従来方式では、階段の上り下りには上手く対応できていないようである。

5.1 節で述べた登りと下りでの階段の差があることが、5.2 節で述べた実験によって確認することができた。しかし、著者らが実験当初に想定した差よりも非常に小さい。この点は、いくつかの理由から気になっている。まず、著者らは、通常通りの気持ちで階段を上り下りして計測した。これは、通常よりも早急な歩みのようであったが、この場合登りと下りで、2, 3 歩の違いが出る。一方、実験では、被験者に対して、3 人の計測者が追走する形で実験を行っている（図 22）。被験者には自然に歩くように指示しているが、実験環境から、被験者は自然な階段の上り下りよりも、大回りでゆっくり歩いているのではないかと想定している。そのため、実験の仕方から、再度検討する必要があると考えている。

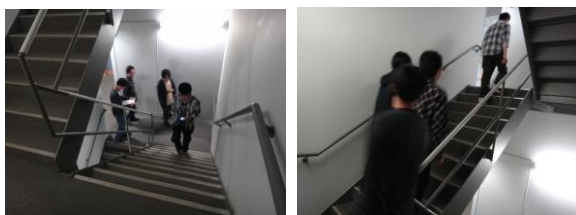


図 22 階段歩行時の実験の様子

6.2 階段の歩数カウント

著者らは、プログラム開発時の検証作業の中で、上り時は、加速度による衝撃が少なくなり、下り時は歩数の間隔が短くなることで、少な目に歩数検知されることを想定していた。しかし実験では、著者らが想定した値よりも、歩数値が正しく数えられている。

6.1 節と同様に、実験の仕方を再検討することで、この実験結果で正しいのか検証作業が必要であると考えている。そして、このような検証作業をすることで、実運用した時にも正しく計測できるようになると考えている。

6.3 ジャイロセンサーによる方位の変化による階段の終了判定

階段は、図 16 に示す構造をしている。この構造を通ったときの一連の動きを判定する、すなわち連続した曲がりのパターンを認識することで、階段の終了判定が可能であると思われる。具体的には、ジャイロセンサーによって得られた波形を積分し、角度に変換する。このときの波形の形に着目することで、階段の終了判定を行えるのではないかと考えている。

6.4 様々な構造の階段

本研究で実装した地図情報のデータ構造では、階段によるフロアー移動をした際、垂直移動する階段でしか、デッドレコニングを行えない。これはたとえば螺旋階段では、この手法でデッドレコニングできるが、階段移動によって水平面の座標が異なる位置に移動する場合の階段では、デッドレコニングできないことを意味している。実際的な階段を考慮したデータ構造に変更する必要がある。ただし、これは地図情報を使うことで、十分対応可能であると考えている。

6.5 「自然な動き」の難しさ

被験者には、自然に歩くこと、および自然に階段を上り下りすることをお願いしているが、通常よりは規則正しい動きをしてしまう傾向がある。これは実験のたびに、判定が成功したか失敗したかが被験者に分かることにより、被験者が成功を意識した動きになるようである。特に、Android 端末を持って実験をするという環境上、実験途中でも、検出の成功・失敗が分かるため、実験の回数を重ねるごとに、成功率が上がる傾向があるようである。

今回、5.3 節で述べたように、階段移動の開始・終了判定を成功させることができた。しかし、総合的な実験は成功していない。これは今回の被験者の歩幅が、3.3 節で実験した時よりも狭く、マップマッチングに成功しないためである。

この原因は、3.3 節で行った実験と 5.3 節で述べた実験年度が異なるため、実験に協力してくれた学生が異なることが考えられる。また、先に述べたように、被験者の慣れにより、3.3 節で行った実験は、本来よりも良い結果が得られているためと考えられる。

自然な動きで実験をするための工夫、または被験者ごとに閾値を自動調整できる機能などが必要であると思われる。

7. おわりに

スマートフォンに内蔵された、加速度センサー、ジャイロセンサーおよび気圧センサーを用いたデッドレコニングにより、大学研究室があるような複数フロアーのデッドレコニングを行えることを確認した。しかし、曲がり判定（3.2.2 項）で示したように、図書館のような曲がり角が頻繁にあるような箇所では、実用に耐えないことが分かっている。

今後は、それぞれの課題を解決していき、実用に向けて更なる改良を進めていく予定である。

謝辞

本研究は文部科学省科学技術研究費基盤(C)課題番号 24500219 の補助を得て行った。

参考文献

- 1) 国立天文台編, 平成 24 年 理科年表, 丸善, 2011, pp 気 156(332)-気 157(333).
- 2) 北川拓, 新井イスマイル. スマートフォン内蔵ジャイロセンサによる屋内方位推定精度向上. 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] 2013-UBI-37(11), 1-8, 2013-03-07 一般社団法人情報処理学会
- 3) 野田浩幸, “準天頂衛星システム技術実証 準天頂高精度測位実験,” GPS/GNSS シンポジウム 2011, pp.27-32, Oct.2011.
- 4) 山田直治, 磯田 佳典, 南 正輝, 森川 博之, “プレゼンスシステム実現のための無線 LAN を用いた高角度な屋内エリア推定方式,” 情報処理学会論文誌, Vol.50, No.8, pp.1845-1855, Aug.2009.
- 5) 椎尾一郎, “RFID を利用したユーザ位置検出システム,” 情報処理学会研究報告, ヒューンインタフェース研究会報告, pp.45-50, May,2000.
- 6) 五百蔵重典, “通信および地図情報が不要なナビゲーションシステム,” DICO2011 シンポジウム, 6C-4, pp.1084-1090, July,2011.
- 7) D. Kamisaka, S. Muramatsu, T. Iwamoto, and H. Yokoyama, “Design and Implementation of Pedestrian Dead Reckoning System on a Mobile Phone,” IEICE TRANS. INF. & SYST., VOL.E94-D, No.6, pp.1137-1146, June,2011.
- 8) 興侶正克, 大隈隆史, 石川智也, 蔵田武志, “装着型内蔵センサモジュールを用いた歩行者の位置・方位計測技術,” 電子情報通信学会誌, Vol.92, No.4, pp.268-275, Apr.2009.
- 9) H. Sunaga, T. Hada, M. Akiyama, S. Ioroi, and H. Tanaka, “Discrimination of Multiple Objects and Expanding Positioning Area for Indoor Positioning Systems using Ultrasonic Sensors,” Aml 2011, LNCS7040, pp.31-40, Amsterdam, The Netherlands, Nov.2011.
- 10) 北陸電気工業株式会社, 3 軸加速度アプリケーションノート, http://www.hdk.co.jp/pdf/AP_Note/anhaam04_v1.01.pdf.
- 11) 須永光, 齊木拓実, 秋山征己, 五百蔵重典, 田中博, 山本富士男: 慣性センサを用いた歩行者の屋内位置推定法に関する基本実験と評価. 電子情報通信学会 ユビキタス・センサネットワーク研究会 (USN) pp. 49-54. 2012.
- 12) 鈴木惇也, 秋山征己, 田中博, 五百蔵重典: スマートフォン内蔵センサを用いた歩行位置推定に関する基本実験と評価. 情報処理学会研究報告, 2012-MBL-64(10), pp.1-7
- 13) 秋山征己, 須永光, 五百蔵重典, 田中博: 超音波センサを用いた広域屋内測位システムの構成と検証実験. 測位航法学会論文誌 Vol. 3, No.3, pp.1-8. 2012
- 14) 村田正秋, 瀬川爾朗, 鳥本秀幸: IMES の技術動向 : シームレス三次元測位・航法の新技术, 電子情報通信学会誌, ISN-09135693, vol. 95, No.2, page.119-124, 2012
- 15) 岡龍太, TranXuanDuc, 新井イスマイル, 西尾信彦: 位置特定インフラ専用無線 LAN アクセスポイントの試作と測位精度制御の検討評価, 情報処理学会 全国大会講演論文集, No.1 pp275-277. 2011
- 16) 暦本純一, 塩野崎敦, 末吉 隆彦: PlaceEngine--実世界集合知に基づく WiFi 位置情報基盤, インタ-ネットコンファレンス論文集, pp95-104, 2006
- 17) 梶克彦, 河口, 信夫: indoor.Locky:UGC を利用した無線 LAN 屋内位置情報基盤, 情報処理学会論文誌, ISSN-03875806, 情報処理学会, Vol.52, No12, pp3263-3273, 2011
- 18) P. Bahl and V.N. Padmanabhan: “RADAR: An In-Building RF-based User Location and Tracking System” . Proceedings of Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies (INFCOM 2000) Vol.2 pp.775-784 (2000)
- 19) 稲村裕之, 飯田康隆, 難波由樹, 屋内位置検知システム用 950MHz 帯無線受信機, 東芝レビュー Vol.66 No.4 pp20-23.2011
- 20) 荻野敦, 恒原克彦, 矢野隆, 渡辺晃司, 加藤猛, 鈴木秀哉, 無線 LAN 統合アクセスシステム(1)-位置検出システムの検討-, 電子情報通信学会総合大会, B-5-203. 仙台, 2003-3, 電子情報通信学会. 2003, p662
- 21) 上野秀樹, 佐藤義之, 片岡淳, 可視光 ID システム, 東芝レビュー Vol.62 No.5 pp44-47.2007
- 22) 秋山征己, 鈴木惇也, 田中博, 五百蔵重典: スマートフォン内蔵ジャイロセンサを用いた曲がり検出手法の提案と評価. 電子情報通信学会総合大会. B-19-45. p.589. 2013
- 23) 鈴木孝幸, 白井宏幸, 森崎雅崇, 田中博: Android 端末内蔵気圧センサを用いた在階推定方法の基礎検討. 電子情報通信学会総合大会. B-19-3. p.578. 2013
- 24) ウェルサポート (商品ページ) . ニプロ株式会社. <http://www.nipro.co.jp/ja/news/2008/document/080423.pdf>. 2013-05-16
- 25) <http://keisan.casio.jp/has10/SpecExec.cgi?id=system/2006/1257609530>
- 26) 小林誠, 黒沼紀彦, 五百蔵重典: QR コードによるナビゲーションシステムのスマートフォンを用いた実装. 情報処理学会研究報告. ITS, 2012-ITS-51(9), 1-8, 2012