

# 歩行・停止情報を用いたユーザの移動経路推定手法

宮下 浩一<sup>1,a)</sup> 寺田 努<sup>2</sup>

受付日 2015年9月28日, 採録日 2016年4月5日

**概要:** 近年, ユーザの日常生活や体験をデジタル化し記録するライフログに関する研究がさかんに行われている. このライフログにおいて重要なコンテキストである位置情報の取得方法は多数存在しているが, 消費電力が大きくウェアラブルコンピューティング環境の有限なバッテリーでは長期間にわたる運用が難しい, 測位精度が低い, 使用できる環境に限られる, などいずれの手法も問題を有している. そこで本稿では, 消費電力の小さな歩行認識デバイスを用いて, ユーザの2状態(歩行中・停止中)のみを認識し, その情報と地図情報をもとにユーザの位置を推定する手法を提案する. 提案手法では, ユーザが停止する地点は交差点であると仮定し, 歩行時間から推定した歩行距離と, 交差点間の距離が等しい経路を求める. 評価実験により, 本手法と既存の位置取得方法を組み合わせることである程度の精度で経路推定が行えることが明らかとなった.

**キーワード:** ウェアラブルコンピューティング, 移動経路推定, 歩行認識デバイス

## A Route Estimation Method Using Walking/Staying Status

KOICHI MIYASHITA<sup>1,a)</sup> TSUTOMU TERADA<sup>2</sup>

Received: September 28, 2015, Accepted: April 5, 2016

**Abstract:** Recently, there has been increasing interest on researches for life-log systems that record the users' daily life and experience. There are several methods to acquire user location that is very important context on life-log systems. However, these methods have problems such as large power consumption, low accuracy, and environmental restriction in the field of wearable computing. Therefore, in this paper, we propose a new method to acquire user location. Our method requires a small and low battery device like pedometer, and estimates user location by matching the recognized user status walking/staying and the map information. The calculation result confirms that our method correctly estimates the route within 10 minutes.

**Keywords:** wearable computing, route estimation method, walking recognition device

### 1. はじめに

近年の情報機器の小型化・軽量化にともない, コンピュータをつねに身につけて生活するウェアラブルコンピューティングが注目を集めている. ウェアラブルコンピューティングでは, さまざまなセンサやコンピュータを身につけることで, ユーザは時間や場所を選ばずにコンピュータを利用できるようになる. ウェアラブルコンピューティン

グにおけるアプリケーションの1つとして, いつ・どこで・何をしたのかなどの, ユーザの日常生活や体験をデジタル化し記録する, ライフログが注目を集めている. ライフログにより, たとえば1日を振り返って日記を書く, 忘れ物・落し物に気付いたときに落とした場所を特定する, 日々の健康状態を管理する, さらに自叙伝を作成するといったことが可能となる.

ライフログにおいて「どこで」にあたる位置情報は重要な情報であり, これまでに多数の位置取得方法が提案されているが, 低消費電力で常時高精度に位置を取得することは難しい. たとえばGPSによる測位では, 測位誤差は約10mとある程度高い精度が得られるが, ウェアラブルコンピューティング環境での常時利用には電力消費が大きい.

<sup>1</sup> 株式会社三菱総合研究所  
Mitsubishi Research Institute, Inc., Chiyoda, Tokyo 100-8141, Japan

<sup>2</sup> 神戸大学大学院工学研究科  
Graduate School of Engineering, Kobe University, Kobe, Hyogo 657-8501, Japan

a) k.miya@mri.co.jp

そこで本稿では、消費電力の小さな歩行認識デバイスを用いて、ユーザの2状態（歩行・停止、以降歩行状態ログと呼ぶ）を認識し、その情報と地図情報をもとにユーザの位置を推定する手法を提案する。提案手法では、ユーザが停止する地点は交差点であると仮定し、歩行時間から推定した歩行距離と、交差点間の距離が等しい経路を求める。また、低頻度で定期的に取得するGPSからの位置情報を用いることで、地図情報をもとにユーザの位置を推定する。本研究では実際にプロトタイプデバイスを実装して評価を行い、その性能を評価する。

本稿の構成は以下のとおりである。2章では本研究に関連するウェアラブルコンピューティングや位置測位の研究を紹介し、3章で筆者らが提案する経路推定手法を説明する。4章でシステム実装について述べ、5章で提案手法の評価を行い、最後に6章で本稿をまとめる。

## 2. 関連研究

### 2.1 センサの消費電力

近年、スマートフォンメーカーは激しい開発競争を行っている。Android 端末においては同一OSを採用する競合他社との差別化のために、ハードウェアの高機能化と低消費電力化、軽量化を進めている。このような背景から、ハードウェアを構成する各センサにおいても、低消費電力化が進められている。代表的なセンサの消費電力は表1のとおりである。

本研究が対象とする測位技術の1つであるGPSは、低消費電力化が進んでいるものの、他のセンサと比較すると依然として消費電力の大きなデバイスである。

このような背景から、GPSの利用を抑える研究がさかんに行われている。加藤らは、消費電力の小さい加速度センサなどを用いて利用者の周囲の状況を把握し位置測位をする必要があるときのみにGPSを起動することで、消費電力の大きいGPSの更新頻度を低減する技術を開発している[7]。中川らは端末に搭載された各種センサを用いて、移動方向や監視対象スポットへの距離に応じて測位回数を削減することで省電力化する手法を提案している[8]。

表1 スマートフォン向けセンサの消費電力

Table 1 Power consumption of smartphones sensors.

センサ	製品例	消費電力
GPS	EVA-M8M <sup>1)</sup>	75mW(25mA @ 3V)
ジャイロセンサ	BMG160 <sup>2)</sup>	15mW(5mA @ 3V)
加速度センサ	BMA355 <sup>3)</sup>	390 $\mu$ W(130 $\mu$ A @ 3V)
電子コンパス	AK09911 <sup>4)</sup>	7.2mW(2.4mA @ 3V)
光度センサ	APDS-9960 <sup>5)</sup>	600 $\mu$ W(200 $\mu$ A @ 3V)
近接センサ	APDS-9960	2370 $\mu$ W(790 $\mu$ A @ 3V)
気圧センサ	HSPPAD038A <sup>6)</sup>	28.5 $\mu$ W(9.5 $\mu$ A @ 3V)

### 2.2 ウェアラブルコンピューティングとライフログ

コンピュータやさまざまなセンサを身に着けるウェアラブルコンピューティングでは、ユーザは時間や場所を選ばずにコンピュータを利用できる。このようなウェアラブルコンピューティングシステムを用いることで、たとえばユーザを目的地まで案内するウォークナビゲーションシステムやユーザの健康状態をつねに監視し、危機を知らせるシステム[9]などユーザの日常生活に密着したサービスが提供できる。

ウェアラブルコンピューティングを利用したアプリケーションに、ユーザの日常生活や体験をデジタル化し記録するライフログがある。これは、カメラで撮影した写真やビデオで撮影した動画、マイクで録音した音声、身に着けたセンサから得られたデータを解析し得られたユーザの状態や行動などのコンテンツを、時間や場所、キーワードなどによりタグ付けして保存することである。

ライフログにおけるコンテンツ生成やタグ付けに関する研究は現在さかんに行われている。たとえば、ユーザが身に着けた各種センサを用いてユーザの状態や行動を認識するコンテキストウェアの分野では、Laerhovenらの加速度センサなどの複数のセンサを用いた行動認識システムPorcupine[10]が知られている。また、堀らは映像コンテンツからコンテキストを推定しタグ付けする手法[11]を提案している。この手法では、ビデオの映像や音声とともに、脳波計から得られた $\alpha$ 波の波形や、モーションセンサから得られたユーザの状態・行動、GPSで取得した位置情報を検索タグとして保存することで、ユーザがライフログビデオを効率良く閲覧・検索できるようにしている。

ライフログを支援するデバイスとしては、スマートフォンのカメラ撮影時に位置情報を記録する機能や、GPS機能連携デジタルカメラなど、写真と撮影した場所を記録する機能を持つデバイスがある。撮影した写真は、たとえば、Google PicasaとGoogle Earthを用いて、地図上に重畳させて表示できる。このように、コンテンツに位置情報を付加して保存することで、ユーザは撮影時の状況を思い出しながら楽しくコンテンツを閲覧でき、また、フォルダ内の写真一覧から写真を探すよりも簡単に目的の写真を見つけることができる。しかし上記のデバイスは、GPS受信機を常時ONにすると消費電力が大きくなり、撮影時のみ電源を入れると測位までに時間を要するためにすぐに撮影できない。

### 2.3 位置取得手法

冒頭で述べたGPS以外にも、位置を取得する手法は多数存在する。河口は、サーバにWiFiのアクセスポイントの電波情報と位置とを関連付けたデータベースを構築し、ユーザが取得できるWiFiの電波から現在位置を推定する手法[12]を提案しており、Locky.jp[13]でサービスを提供し

ている。類似するサービスに、Place Engine [14] や Google My Location [15] などがある。これらの手法は GPS の電波が取得できない地下などでも位置を測位できるが、サーバに登録されたアクセスポイントの位置情報に精度が依存し、測位精度は 5-30m 程度とばらつきが大きい。また無線 LAN による通信も電力消費が大きい。

RFID を用いた位置の推定に関する研究もさかんに行われている。椎尾は Passive 型 RFID を用いたユーザ位置検出システムを提案している。彼らは RFID リーダを着けた履物と、RFID タグを敷き詰めたカーペットを用いて、室内での高精度な位置検出システム [16] を提案している。また、Active 型 RFID を用いた手法は、川島らの実証実験 [17] により誤差 2.5m と高い測位精度が得られることが分かっている。しかし、RFID の通信範囲エリアは、Passive 型で数 m、Active 型でも十数 m と狭いため、環境に多数の RFID タグを埋め込む必要がある。

また、ウェアラブルコンピューティングにおける位置推定手法には、3次元位置姿勢センサである磁気トラッカと姿勢センサであるジャイロを身に着けた環境において、ユーザの歩行軌跡を求める濱口らの手法 [18] や、室内に配置した画像マーカをユーザの身に着けたカメラで撮影し、マーカとのタグ付けされた位置情報からユーザの位置を特定する羽原らの手法 [19] などがある。前者はセンサから得られるデータの誤差が蓄積されてしまい、長時間の運用には向かない。後者は、画像マーカを設置した環境に利用範囲が限定されるという問題点を有している。

近年では GPS や無線 LAN のほかに、Bluetooth を用いた位置取得手法が登場している。2010 年に策定した BLE (Bluetooth Low Energy) を用いた屋内位置測位では、BLE 受信端末が屋内に設置された複数の BLE 発信機の信号を受信して、従来よりも低い消費電力で屋内位置を特定することが可能になると期待されている。石塚らは約 50 名の被験者に対して展示会場内に複数設置された BLE 発信機の信号の受信状況を測定する基礎実験を実施し、測位精度を評価した [20]。また、田村らの研究では、ポジショニング範囲を限定するために、Bluetooth を従来の無線 LAN を用いたポジショニングシステムに組み合わせることで位置決定精度の向上を図る実験を行った。その結果、測位誤差を 7.56m から 2.62m へと小さくすることが可能になった [21]。

さらに従来の GPS では電波が届かない、地下や屋内での測位を可能にする手法として、IMES が提案されている。Dinesh らは IMES を用いて地下街における被験者の位置を取得し、経路誘導を行う実証実験を行い、シームレスで高精度な屋内測位を行えることを確認した [22]。

### 3. 提案手法

#### 3.1 想定環境

想定環境として、ユーザはその状態 (歩行・停止) の変化をタイムスタンプとして記録できるデバイスを身に着けているとする。たとえば、石川らが提案する加速度センサから歩数を認識するデバイス [23] や、歩数計に時刻保存機能を加えたようなものである。我々は省電力性の観点から、図 1 に示すような歩行認識デバイスを想定している。このデバイスは、図の左に位置する筒内の球が、ユーザが歩くたびに振動する。停止中は球が筒の床面と接地しており電流が流れるが、震動すると床から離れるため回路が断線する。この断線したタイミングをマイコンが記録しておくことで、初めに想定した歩行・停止のタイムスタンプを記録できる。このような機構は市販の歩数計など同様に省電力であり、ボタン電池で 1 年間駆動させるなど十分な省電力性を持つといえる。本研究ではこのようなデバイスから得られる歩行開始時刻と停止時刻をもとに歩行経路の推定を行う。また、提案するデバイスのみで歩行経路を予測するだけでなく、GPS のような位置取得デバイスを用いて、消費電力に大きな影響を与えない測位間隔 (数分に 1 度単位の低頻度) で位置を記録し、測位した点間の経路を、歩行状態ログをもとに選定することも想定している。

想定するユーザの歩行パターンについては次のとおりとする。伊藤らの研究によると、歩行者信号が赤表示で待ち時間が 0 秒~60 秒の歩行者 (全体のおよそ 9 割) は、当該信号で信号待ちを行うものの、60 秒以上待ち時間がある場合には別の交差点でわたるために進路を変更するなどの要因で当該信号の信号待ちを行わないことが多いことが示されている [24]。また森らの研究によると、自由散策行動において 5 分間で 1 人 1.5 回のアクションをとり、アクションのうち経路の折り返しのアクションが発生した割合は 1.8%、立ち止まりのアクションが発生した割合は 6.8% であった [25]。これらの歩行者の行動に関する先行研究をふまえて、想定するユーザの歩行として、ユーザは交通法規を遵守し、交差点で赤信号であれば信号待ちをし、横断歩道以外で道路を横断することはないものとする。また、歩行

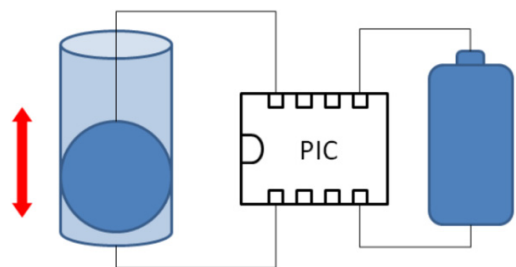


図 1 歩行・停止記録デバイスのイメージ

Fig. 1 The conceptual image of the wearable device for walking detection.

中に立ち止まったり、走りだしたり、来た道を折り返したりすることはない歩行を想定する。すなわち、ユーザはある程度同じ速度で歩行し、ユーザが停止する地点は交差点であると想定する。

本手法では道路網をネットワークに見立て、交差点をノード、交差点間の道路をリンクとして考える。つまり、あるリンクはそのリンクが持つノード以外では他のリンクと接続しない。

上記の環境下で、以下のようなサービスシナリオを想定している。

- 落とし物をしたことに数時間後に気付いたユーザが、落とした位置を知るためにライフログ中の位置情報を参照するも、GPSの測位間隔は長いいため飛び飛びの位置情報しか得られない。そこで、提案手法を用いることで、測位地点の間を結ぶ経路が分かり、ユーザは落とし物を探しやすくなる。
- ユーザがウェアラブルカメラを身に着けて、動画を撮影しながら散歩をした。家に帰ったユーザは動画の中から散歩の途中で印象的だった場面のキャプチャ画像の整理をするために、地図上に画像をマッピングしようとするが、画像の枚数が多く手間である。そこで、提案手法で得られた経路上に、画像を撮影した時刻から撮影場所を推測して地図上に提示することで、ユーザの画像管理を助ける。
- 健康促進のために毎日欠かさず散歩をするユーザが、歩行認識デバイスを用いることで、散歩中の歩数だけでなく、歩いた軌跡も地図上に表示されるようになり、ユーザの散歩へのモチベーションが維持される。

### 3.2 測位情報を持たない場合の経路推定手法

本研究では、以降、歩行開始地点から目的地まで歩くことを総歩行、歩き始めてから交差点で停止するまでを部分歩行と呼ぶ。総歩行はいくつかの部分歩行で構成される。歩行ログには、部分歩行の開始時刻と終了時刻の組が複数個記録されている。また、ここでは総歩行を行っている間はGPSなどによる測位情報は得られないと仮定する。

提案手法では、歩行ログと歩行開始のノードから経路の推定を行う。初めの部分歩行に対し、歩行速度  $v$  と部分歩行の歩行時間  $t$  から、ユーザの歩行距離  $v \times t$  を求める。歩行距離には歩行速度の変化を見込みマージン  $m$  を持たせ、歩行開始ノードからのリンク長が最短歩行距離 ( $v \times t \times (1 - m)$ ) から最長歩行距離 ( $v \times t \times (1 + m)$ ) の範囲に入る経路を探し、その終点ノードを保持する。たとえば、図2のような、縦20m横30mの格子からなる地図において、ノード1を歩行開始ノード、ユーザの歩行距離  $v \times t = 100\text{m}$ 、マージンを0.1とした場合、ノード1からの経路長が90~110mの範囲にある、9個の赤色ノードが終点ノードの候補として選択される。

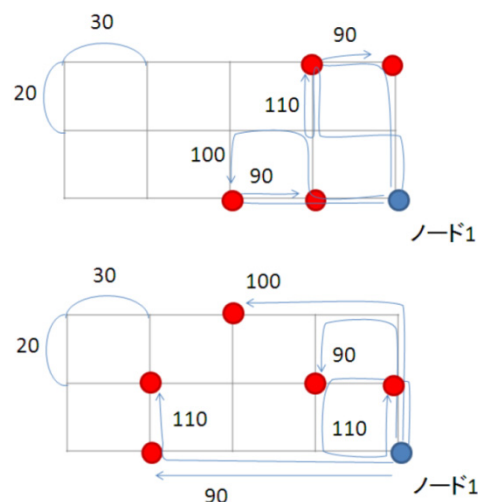


図2 終点ノード候補 (見やすさのため2つに分けて表示)  
Fig. 2 Finding the node at the end of the partial pathway.

さらに、1つ前の部分歩行の終点ノードを次の部分歩行の開始ノードとして、新たに経路を探索する。この操作をすべての部分歩行に対して行い、総合的な歩行経路を導出する。

#### 3.2.1 経路のフィルタリング

上記の手法を単純に適用した場合、10分程度の歩行でも、候補となるノードが膨大になり現実的な解が得られない。これは図2からも分かるように、候補の経路には折り返しや回り道まで含まれてしまうからである。そこで、候補となるノードを絞るためにフィルタを適用する。提案するフィルタは、部分歩行内で用いるものと、歩行全体で用いるものに分けられる。

##### (1) 部分歩行で用いるフィルタ

- 折り返し防止フィルタ：部分歩行内で以前通過したノードへ再び戻ってくる経路を削除することで、折り返しや巡回する歩行を除外する。
- 始点終点長フィルタ：不自然な回り道をしている経路を削除する。部分歩行の始点ノードと終点ノードの直線距離と経路長とを比較する。具体的には、 $(\text{始点終点長} \times \beta) < (\text{経路長})$  となる経路は候補から削除する。 $\beta$ は定数で、経路長が始点終点長の $\beta$ 倍以上であれば候補から削除される。
- 最短経路長フィルタ：不自然な回り道をしている経路を削除する。部分歩行の始点ノードと終点ノードを結ぶ最短経路長と比較する。 $(\text{最短経路長} \times \gamma) < (\text{経路長})$  となる経路は削除する。 $\gamma$ は定数で、経路長が最短経路長の $\gamma$ 倍以上であれば候補から削除される。

##### (2) 歩行全体で用いるフィルタ

- 直線距離フィルタ：歩行開始ノードから、停止候補の交差点ノードまでの直線距離と、歩行時間  $t$  と歩行速度  $v$  から求めた歩行距離とを比較する。 $(\text{直線距離}) < (t \times v \times \delta)$  となる経路は削除する。 $\delta$

は定数で、直線距離が歩行距離  $t \times v$  の  $\delta$  倍を下回る場合、歩行開始点からあまり離れていないと見なし削除する。

### 3.2.2 経路のスコアリング

上記までの手法を適用して得られた経路候補に対し、歩行速度の変化をもとに経路のスコアリングを行う。  $K$  個の部分歩行からなる総歩行において、各部分歩行の歩行時間を  $t_1, t_2, \dots, t_k, \dots, t_K$ , 歩行経路を  $l_1, l_2, \dots, l_k, \dots, l_K$  と表現した場合、経路のスコアリングは、総歩行時間に対する各歩行時間の割合と総歩行距離に対する各歩行距離の割合の差 (式 (1)) とする。提案手法においては、この値が小さいほど歩行距離と歩行時間とが比例している経路であるため、正しい経路であると仮定する。

$$\text{歩行時間と歩行距離の比の差 (スコア)} = \sum_{k=1}^K \left| \frac{t_k}{t} - \frac{l_k}{l} \right| \quad (1)$$

たとえば、ある総歩行において 2 回の停止により 3 個の歩行時間 (40 秒, 80 秒, 100 秒) が与えられ、提案手法により 2 つの経路 (経路 1 (50 m, 110 m, 130 m), 経路 2 (60 m, 110 m, 140 m)) が導出される場合、経路 1 のスコアは 0.0313 (式 (2)), 経路 2 のスコアは 0.0235 (式 (3)) となり、この例では経路 2 が選択される。

$$\begin{aligned} \text{経路 1 のスコア} &= \left| \frac{40}{220} - \frac{50}{290} \right| + \left| \frac{80}{220} - \frac{110}{290} \right| \\ &+ \left| \frac{100}{220} - \frac{130}{290} \right| = 0.0313 \end{aligned} \quad (2)$$

$$\begin{aligned} \text{経路 2 のスコア} &= \left| \frac{40}{220} - \frac{60}{310} \right| + \left| \frac{80}{220} - \frac{110}{310} \right| \\ &+ \left| \frac{100}{220} - \frac{140}{310} \right| = 0.0235 \end{aligned} \quad (3)$$

### 3.3 GPS などの測位デバイスを用いた場合の処理

3.2 節で提案する手法は、犬の散歩で自宅近辺を周回するなど歩行開始地点と目的地が同一の場合に、直線距離フィルタなどの影響で正しく経路を推定できないという課題や、移動時間が長くなるにつれて推定精度が低下するという課題が想定される。

そこで本節では、GPS のような位置取得デバイスを用いて、消費電力に大きな影響を与えない測位間隔 (数分に 1 度単位の低頻度) で位置を記録し、測位した点間の経路を、歩行状態ログをもとに選定する手法を提案する。一定間隔で測位できると仮定した場合の経路推定は次の手順で行う。

- (1) 測位情報に基づく始点・終点・歩行エリアの決定
- (2) 歩行エリア内での経路検索
- (3) 経路のフィルタリングおよびスコアリング

以下、各ステップにおける処理を説明する。

- (1) 測位情報に基づく始点・終点・歩行エリアの決定

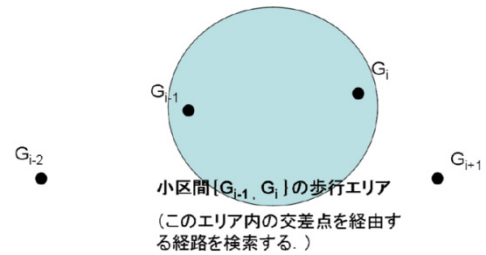


図 3 歩行エリアの設定

Fig. 3 The walking area of a section is determined by GPS points.

本研究では、一定間隔 (数分~数十分程度) で取得した位置情報の活用を前提としている。GPS 点列 ( $G_0 \sim G_n$ ) を、小区間  $\{G_0, G_1\}, \{G_1, G_2\}, \dots, \{G_{i-1}, G_i\}, \dots, \{G_{n-1}, G_n\}$  に分割し、小区間  $\{G_{i-1}, G_i\}$  において、始点  $G_{i-1}$  と終点  $G_i$  を結ぶ経路を推定する。経路の推定にあたって、歩行者のとりうる歩行エリアを設定する。一般的な歩行者は、小区間において大きな回り道を行わないと仮定し、小区間の、始点  $G_{i-1}$ , 終点  $G_i$  を直径とする円を若干大きくした円 (評価時には始点終点間の距離を 1.2 倍した値を利用) を歩行エリアに設定する (図 3 参照)。

#### (2) 歩行エリア内での経路検索

歩行エリア内において、始点  $G_{i-1}$ , 終点  $G_i$  を結ぶ経路を想定しうる限り検索する。ただし、歩行エリア内における折り返しや回り道まで考慮すると経路が発散するため、小区間内では折り返しや回り道は発生しないと仮定する。具体的には、始点  $G_{i-1}$  と終点  $G_i$  を結ぶ、複数の交差点ノードからなる経路  $\{G_{i-1}, N_1, N_2, \dots, N_{j-1}, N_j, \dots, G_i\}$  ( $N_j$  は走行エリアに含まれる任意の交差点ノード) は、必ず終点へ近づく交差点ノードを経由する。すなわち、距離  $(N_{j-1}, G_i) > \text{距離}(N_j, G_i)$  を満たすものとする。

#### (3) 経路のフィルタリングおよびスコアリング

得られた経路を対象に、歩行・停止の状態を記録した歩行状態ログを利用して、3.2 節で提案した手法を始点と終点が固定されていると見なし経路の絞り込みを行う。

ただし、3.2 節の経路推定手法における歩行速度  $v$  は GPS の測位データをもとに動的に決定する。具体的には、小区間における始点  $G_{i-1}$  と終点  $G_i$  を結ぶ最短経路を算出し、最短経路長を小区間における歩行時間の合計で除算して導出する。実際の歩行経路は、必ずしも最短経路を選択しているとは限らないが、前ステップで仮定したように、数分程度の歩行時間である小区間内では、大きな回り道をとることは一般に考えにくく、最短経路に近い距離の経路を選択していると仮定した。

上記の過程で求めた歩行速度  $v$  と歩行時間  $t$  を用いて経路候補の導出とスコアリングを行い、小区間内における経路を導出する。これらの処理をすべての小区間に対して行い、最終的な経路を得る。

本手法により、GPS 測位点間で立ち止まることがあれば、単純に歩行時間  $t \times$  歩行速度  $v$  で求められる想定移動距離に最も近い経路を採用する手法よりも経路推定精度が向上することが期待される。

#### 4. 実装

提案システムのプロトタイプを実装した。歩行・停止情報の取得デバイスとして、市販の万歩計のケースに入るサイズの歩行情報取得デバイスを作成した。実装したデバイスを図 4 に示す。このデバイスは、アトメル社のマイコン



図 4 実装したプロトタイプデバイスの外観  
Fig. 4 Prototype device.

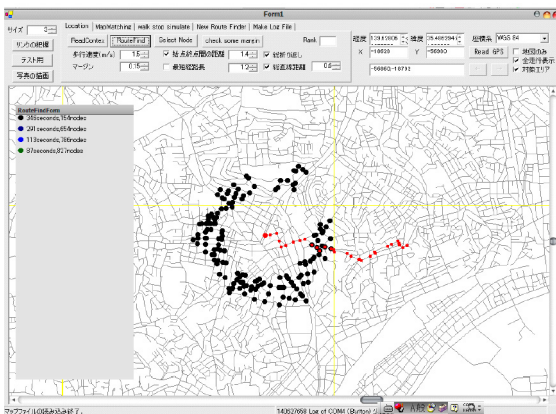


図 5 アプリケーションの例  
Fig. 5 Example of Application.

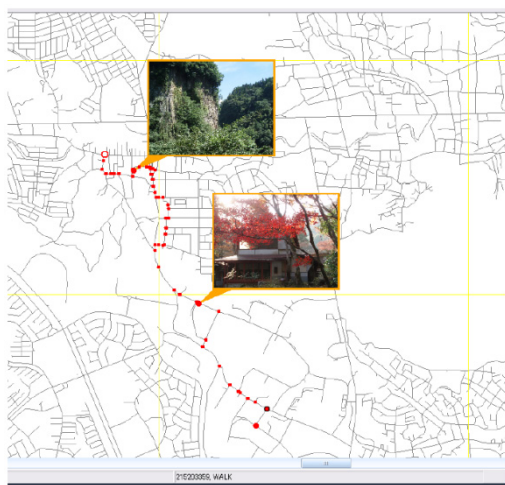


図 6 画像表示機能の実行例  
Fig. 6 Example of picture view function.

ATMEGA168-10AI を搭載しており、各歩行のタイムスタンプを EEPROM に書き込む機能を持つ。デバイスと PC は RS232C を通して通信し、消費電力低減のため、デバイス上の RS232C ドライバはデバイスの電源投入後約 20 秒間だけ動作を許可し、この間に通信を検出したら動作し続けるようにした。万歩計のセンサとしては、市販の歩数計に搭載されているセンサをそのまま用いた。これは、リードスイッチと磁石を先端につけた振り子で構成されており、歩行によりリードスイッチに磁石が近づいたり離れたりでパルス信号が出ることを利用している。マイコンは低消費電力モードで 1 秒ごとにカウンタを回しているため、タイムスタンプの解像度は 1 秒である。

ソフトウェアとしては、デバイスから読み込んだデータをもとに経路を取得するシステムを実装した。地図データとしては国土地理院の発行する 1/2500 数値地図を利用している。システムの動作画面を図 5 に示す。このアプリケーションでは、各種のパラメータを変更しながら目的地候補の点集合を図示したり、スコア順に経路を提示したりするといった機能を持つ。また、アプリケーション例として、図 6 に示すように、動画のキャプチャ画像の時刻情報をもとに提案手法で位置を推定し、歩行経路上に画像を配置するアプリケーションを作成した。

#### 5. 評価

##### 5.1 補助デバイスなしの場合の評価

ユーザの日常的な歩行 15 件（平均歩行時間 13 分、主な歩行は大阪府吹田市エリア）を対象に経路探索を行った。地図情報としては国土地理院の発行する 1/2500 の数値地図を用いた。また、ユーザの歩行速度は一定の値 (1.5 m/s) とした。これは一般的な成人男性の歩行速度である。

フィルタを 1 つも用いずに経路の探索を行った場合、ただかだか 10 分間の歩行にもかかわらず、候補リンクの数が膨大になり発散した。フィルタを用いない場合、折り返しなどを許してしまうため、短い道路リンクがあると、1 回だけ折り返す・2 回折り返す・折り返さないなどのように何通りもの経路が生成できてしまうためである。

日常の歩行においては折り返しや巡回などは、少なくとも歩き始めてから 1 度止まるまでの部分歩行においてはほとんど起こらないと考えられるため、以降では、部分歩行における折り返しフィルタはつねに使用するものとする。

##### 5.1.1 歩行速度のマージンと終点候補数の関係

歩行速度のマージン  $m$  を変化させ、終点ノードの候補の数を確認した。結果を表 2 に示す。ただし、候補となるノードが膨大で発散した場合は、N/A としている。また、“( )” 内の数字は、スコアリングした結果をソートして得られた正解ノードの順位である。“-” は正解ノードが終点候補のノード群に含まれていなかったことを意味する。

歩行 9, 10 では、終点候補のノード数が膨大になり、正

表 2 マージンと終点候補のノード数

Table 2 Margins and the number of nodes at the end of the partial pathway.

試行	マージン				総歩行時間 (秒)	停止回数
	0.05	0.10	0.15	0.20		
1	57(27)	85(49)	99(46)	111(55)	727	1
2	260(115)	292(187)	318(188)	332(214)	736	0
3	234(64)	271(99)	315(61)	362(90)	437	0
4	763(685)	950(664)	N/A	N/A	977	2
5	414(195)	486(85)	513(234)	549(345)	590	1
6	291(-)	568(-)	768(760)	913(809)	938	3
7	321(224)	619(406)	648(341)	692(127)	716	2
8	58(-)	63(-)	75(63)	80(64)	578	1
9	202(160)	316(240)	N/A	N/A	1244	1
10	N/A	N/A	N/A	N/A	1189	1
11	349(295)	517(408)	592(469)	769(413)	1058	2
12	13(-)	35(-)	59(44)	72(57)	772	2
13	318(-)	376(361)	581(201)	622(74)	681	3
14	7(-)	21(6)	23(9)	26(12)	521	1
15	355(20)	439(321)	489(35)	568(129)	847	0

表 3 速度と終点候補のノード数

Table 3 Velocity and the number of nodes at the end of the partial pathway.

試行	速度 (m/s)	マージン			
		0.05	0.10	0.15	0.20
6	1.4	210(-)	421(-)	570(-)	813(-)
	1.5	291(-)	568(-)	768(760)	913(809)
	1.6	426(-)	725(705)	964(716)	1051(446)
8	1.4	17(-)	38(-)	53(-)	72(-)
	1.5	58(-)	63(-)	75(63)	80(64)
	1.6	38(-)	59(2)	70(16)	77(51)
12	1.4	65(-)	148(-)	243(-)	342(-)
	1.5	13(-)	35(-)	59(44)	72(57)
	1.6	144(-)	284(-)	395(-)	465(331)

しく終点候補を導くことができなかつた。この2歩行は、歩行時間が20分を超える歩行であり、かつ、住宅地や市街地を歩行していたため、条件を満たす経路が無数に存在していると考えられる。以上から、20分程度の歩行に対しては、単純に提案手法を用いることができないといえる。

歩行6, 8, 12では、マージンが0.05や0.10の場合、正解ノードが終点候補のノード群に含まれていないことから、設定速度と実際の歩行速度が違うことが想定される。そこで、速度を変化させた場合の、終点ノードの候補リンク数について調べた。結果を表3に示す。

ユーザの歩行速度を1.6m/s、マージンを1.0とした場合、試行6, 8においては正解ノードを発見できたことから、設定速度(1.5m/s)より実際の歩行速度が速かったと考えられる。

一方、試行12では、設定速度を1.4m/s, 1.6m/sとした

表 4 フィルタと終点候補のノード数

Table 4 Filter and the number of nodes at the end of the partial pathway.

試行	フィルタなし	フィルタあり			総歩行時間 (秒)	停止回数
		始点終点長	最短経路長	直線距離		
1	99(46)	29(3)	49(5)	37(2)	727	1
2	318(188)	59(59)	247(149)	101(79)	736	0
3	315(61)	167(30)	227(77)	184(38)	437	0
4	1141(-)	1229(935)	1340(1085)	905(-)	977	2
5	513(234)	478(65)	456(44)	267(77)	590	1
6	768(760)	779(775)	978(976)	590(584)	938	3
7	648(341)	247(85)	642(359)	307(150)	716	2
8	75(63)	37(16)	70(26)	23(7)	578	1
9	N/A	N/A	N/A	N/A	1244	1
10	N/A	N/A	N/A	N/A	1189	1
11	592(469)	510(-)	653(538)	292(200)	1058	2
12	59(44)	12(5)	31(19)	17(9)	772	2
13	581(20)	505(63)	574(87)	287(39)	681	3
14	23(9)	3(1)	6(1)	6(1)	521	1
15	489(35)	172(-)	332(-)	219(-)	847	0

場合、マージンを0.15としても正解ノードを発見できず、設定速度を1.5m/sとした場合、マージンを0.15とすると正解ノードを発見できたことから、歩行中の速度は平均するとおおよそ1.5m/sであったが、途中でユーザの歩行速度が変化したと考えられる。

### 5.1.2 フィルタと終点候補数の関係

次に、提案手法で述べた各種フィルタを用いた場合の、最終的な候補のノード数を調べた。マージンは先の評価より0.15とした。始点終点長フィルタにおける $\beta$ は1.4、最短経路長フィルタにおける $\gamma$ は1.2、直線距離フィルタにおける $\delta$ は0.6とした。結果を表4に示す。

結果より、フィルタを用いることで、終点の候補となるノードが絞られていることが分かる。一方で、候補を絞りすぎてしまい、正解ノードまで候補から外れてしまっている場合もある。理想的なフィルタは、正解ノードは残しつつ候補となるノードを減らすフィルタであるが、フィルタを単独で使用した場合、いずれの手法も一定の効果が得られており、どれか1つが優れているというわけではない。

フィルタなしでは、候補となるノードが膨大になっていた走行ログ9, 10は、どのフィルタを用いても、フィルタなしと同様、発散してしまつた。この2つの歩行はともに、周辺に道路がたくさんあるような住宅地や市街地を20分近く歩行している場合である。このような長時間の歩行への対処法としてはマージンを小さくすることが考えられる。マージンが10%から15%へ1.5倍増加すると、該当するエリアの面積は(1.5)<sup>2</sup>倍になり、該当するノード数は面積にある程度比例すると考えられることから、マージンを小さく設定することが候補数の削減に有効であるといえる。

表 5 フィルタの組合せと終点候補ノード数

Table 5 Combination of filter and the number of nodes at the end of the partial pathway.

試行	始点終点長 +最短経路 長	始点終点 長+直線 距離	最短経路 長+直線 距離	全て	総歩 行時 間 (秒)	停 止 回 数
1	29(3)	26(2)	34(2)	26(2)	727	1
2	58(27)	59(30)	98(39)	58(27)	736	0
3	165(121)	167(123)	181(9)	165(121)	437	0
4	1231(956)	750(532)	885(618)	749(533)	977	2
5	363(75)	239(61)	258(67)	228(58)	590	1
6	1223(1214)	669(660)	763(731)	656(647)	938	3
7	556(382)	225(37)	312(166)	225(144)	716	2
8	37(16)	14(5)	21(7)	14(5)	578	1
9	N/A	N/A	N/A	N/A	1244	1
10	N/A	N/A	N/A	N/A	1189	1
11	524(115)	223(42)	296(40)	228(22)	1058	2
12	264(183)	217(149)	254(-)	222(152)	772	2
13	549(137)	265(29)	288(32)	264(26)	681	3
14	14(4)	4(1)	6(2)	3(1)	521	1
15	172(-)	172(-)	216(-)	172(-)	847	0

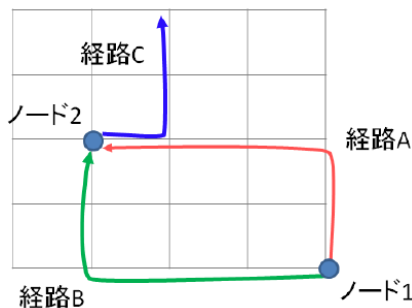


図 7 フィルタの利用によるノード数の増加  
Fig. 7 Increase of nodes by use of filter.

5.1.3 フィルタの組合せと終点候補数の関係

提案手法で述べた各種フィルタを組み合わせで用いた場合の、最終的な候補のノード数を調べた。マージンは先の評価より 0.15 とした。始点終点長フィルタにおける  $\beta$  は 1.4, 最短経路長フィルタにおける  $\gamma$  は 1.2, 直線距離フィルタにおける  $\delta$  は 0.6 とした。結果を表 5 に示す。

フィルタを組み合わせで用いることで、全体的に、候補となる終点ノードの数が絞れていることが分かる。また、始点終点長フィルタと最短経路長フィルタを組み合わせの場合、終点ノードの候補の数が、他の場合と比べて多くなっている。これは、始点終点長フィルタと最短経路長フィルタはともに部分歩行に対して用いるフィルタであるため、組み合わせでも効果が薄くなったためであると考えられる。

また、試行 6 のように、フィルタを単体で用いるよりも組み合わせで用いた方が、終点ノード数が増える場合がある。この理由を図 7 を用いて説明する。

たとえば、フィルタを単独で用いた場合に、ある部分歩

表 6 GPS 測位間隔と一致率の関係

Table 6 Relation between GPS measurement interval and concordance rate.

測位間隔	一致率
3分	64.2%
6分	61.6%
9分	60.7%

行でノード 1 からノード 2 へ行く経路が 2 つ見つかった場合、次の部分歩行においては、折り返し防止フィルタより、経路 C は考慮されない。一方、フィルタを組み合わせで用いた場合、初めの部分歩行で経路 B のみが見つかった場合、次の部分歩行では経路 C まで考慮されてしまう。このような理由から、フィルタを用いることで、逆に候補数が増加する結果となる場合もあることが分かった。

これらを総合的に考えると、すべてのフィルタを組み合わせの場合が最も効率良く候補を絞り込んでいるが、部分歩行でのフィルタと歩行全体で用いるフィルタを各々 1 つ用いた始点終点長フィルタと直線距離フィルタの組合せと比べ大きな差がないことから、後者で十分であるといえる。

5.2 GPS と併用した場合の評価

日常的生活における、21 件の歩行を対象に評価を行った。正解データとして被験者の歩行を毎秒 GPS で記録し、このデータをもとに、歩行・停止の状態の記録と、一定間隔で GPS を利用して測位した場合の位置情報の記録を作成した。歩行した地域は、東京や神奈川などの都心部で道路のネットワーク構造が密な地域である。地図情報には、国土地理院の発行する 1/2500 の数値地図を用いた。また、評価指標としては、推定経路と正解経路とが有する交差点ノードの一致率を用いた (式 (4))。

一致率

$$= \frac{N \left( \begin{matrix} \text{推定経路に含まれるノード} & \cap & \text{正解経路に含まれるノード} \end{matrix} \right)}{N(\text{正解経路に含まれるノード})} \quad (4)$$

まず、歩行速度を一定値 (1.5 m/s)、歩行速度のマージンを 0.15 とした場合の GPS の測位間隔による一致率の違いを表 6 に示す。

測位間隔を延ばせば一致率も低下しているが、その低下量はそれほど大きくないことが分かる。したがって、測位により上昇する精度は、測位した地点近辺のみになっている可能性があり、そもそも歩行速度が現実とずれているといったような場合、測位をしてもそれほど劇的には精度が改善されないことが分かる。

次に、歩行速度のマージンを 0.15、GPS の測位間隔を 9 分とした場合の、歩行速度による一致率の違いを表 7 に示す。

結果より、提案する歩行速度推定アルゴリズムを適用す



表 7 歩行速度と一致率の関係

Table 7 Relation between walking speed and concordance rate.

速度	一致率
動的	67.7%
1.3m/s	65.5%
1.5m/s	60.7%
1.7m/s	41.3%

表 8 マージンと一致率の関係

Table 8 Relation between the margin and concordance rate.

マージン	一致率
0.05	39.2%
0.1	57.5%
0.15	60.7%
0.2	62.0%

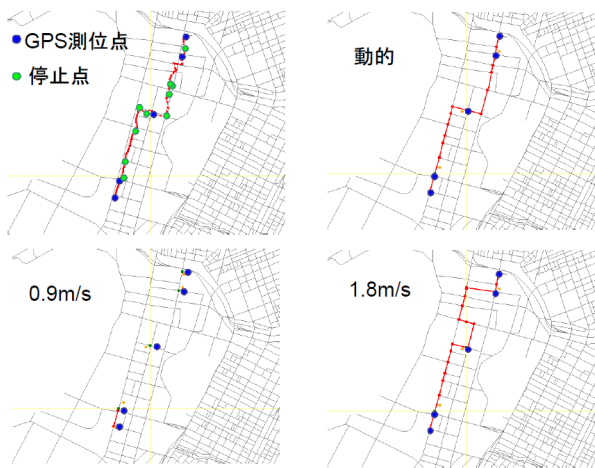


図 8 歩行速度を変化させた場合のマッチング例

Fig. 8 The differences between the correct route and the estimated route when the walking speed was changed.

ることで結果が良くなっていることが分かる。また、速度があまりに現実と異なる 1.7 m/s では、結果が大きく悪化していることが分かる。

歩行速度を一定値 (1.5 m/s)、GPS の測位間隔を 9 分とした場合の、速度のマージンによる一致率の違いを表 8 に示す。

結果から、マージンをあまりに狭くすると一致率が悪化することが分かる。

これらの結果より、提案手法では歩行速度を精度良く推定することが重要であることが分かる。図 8 に示す具体的なマッチング例をあげて説明すると、図左上に示すように、9 分間隔で青い点において GPS 測位が行われ、デバイスにより緑の点で停止点が検出された場合、動的に速度を設定した場合 (図右上) はほぼ正解の経路が得られているが、遅すぎる場合 (図左下) はほとんど経路候補が得られないため総合的な経路が構築できず、速すぎる場合 (図右下) は逆に回り道を多く含む経路が選択されてしまっている。

現状でも 6 割程度の一致率を持っているため、ユーザにアプリケーション上で移動したルートを変更できる機能などを用いれば十分に実用的な推定精度であるといえる。一方、静止・歩行情報だけでなく、万歩計の各歩行のタイムスタンプからより詳細な歩行情報を得ることで、大きく精度を向上させることができる可能性がある。詳細な歩行情報を用いた経路推定手法の提案は今後の課題である。

## 6. まとめ

本研究では、歩行認識デバイスを用いた位置取得手法を提案した。提案手法は、ユーザが歩いていた時間から、ユーザの移動距離を算出し、移動距離に近いルートを探る処理を行い、フィルタで現実的でない経路を取り除き、終点ノードの候補を得るものである。評価の結果、10 分を超える走行では、候補となるノードの数が膨大になったことから、提案するシステムを実現するためには、少なくとも 10 分程度に 1 回の割合で、GPS などを用いて位置情報を更新する必要があることが分かった。また、部分歩行と歩行全体を考慮したフィルタを組み合わせることで、効率良く候補を絞り込むことが可能だと分かった。

今後の課題としては、実運用を通して多くの実データを収集し、より有効なフィルタやスコアリング手法を提案することがあげられる。また、評価の章で述べたように、歩数計から得られる詳細な情報を活用してより精度の高い方式を提案することも大きな課題である。さらに、写真撮影やメールの着信・返信などのイベントが発生し、歩行中に交差点以外の箇所でも停止する場合の対応方策についても今後の検討課題である。

## 参考文献

- [1] EVA-M8M, u-blox, available from <http://www.u-blox.com/>
- [2] BMG160, Bosch Sensortec, available from <https://www.bosch-sensortec.com/>
- [3] BMA355, Bosch Sensortec, available from <https://www.bosch-sensortec.com/>
- [4] AK09911 Magnetometer, ASAHI KASEI MICRODEVICES CORPORATION, available from <http://www.akm.com/>
- [5] APDS-9960, AVAGO TECHNOLOGIES, available from <http://www.avagotech.co.jp/>
- [6] HSPAD038A, ALPS, available from <http://www.alps.com/>
- [7] 加藤大智, 竹腰昇太, 大野雄基, 鈴木秀和, 旭 健作, 渡邊 晃: TLIFES における省電力化を目的とした位置測位手法の提案と実装, 情報処理学会研究報告, Vol.2013, No.13, pp.1–6 (2013).
- [8] 中川智尋, 山田 渉, 土井千章, 稲村 浩, 太田 賢, 鈴木 誠, 森川博之: ジオフェンシングサービスのための間隔可変測位による省電力入圏検出方式の評価, マルチメディア, 分散, 協調とモバイル (DICOMO2013) シンポジウム論文集, pp.1116–1122 (2013).
- [9] Ouchi, K., Suzuki, T. and Doi, M.: Life-Minder: A wearable Healthcare Assistant, *Proc. IWSAWC 2002*,

- pp.791-792 (July 2002).
- [10] Laerhoven, K. and Gellersen, H.: Spine versus Porcupine: A Study in Distributed Wearable Activity Recognition, *Proc. ISWC 2004*, pp.142-149 (2004).
  - [11] 堀 鉄郎, 相澤清晴: ライフログビデオのためのコンテンツ推定, 情報処理学会研究報告, Vol.2003, No.125, pp.157-162 (2003).
  - [12] 河口信夫: 無線 LAN を用いた位置推定とその応用, 電子情報通信学会技術研究報告 (ITS), Vol.107, No.161, pp.1-4 (2007).
  - [13] Locky.jp, available from (<http://locky.jp/>)
  - [14] PlaceEngine, available from (<http://www.placeengine.com/>)
  - [15] Google, available from (<http://www.google.com/gmm>)
  - [16] 椎尾一郎: RFID を利用したユーザ位置検出システム, 情報処理学会研究会報告 (00-HI-88), pp.45-50 (May 2000).
  - [17] 川島和也, 貝沼達也, 岩尾忠重, 藤野信次: ActiveRFID を用いた位置検出方式及び実証実験, 電子情報通信学会技術研究報告 (IN), Vol.103, No.691, pp.49-54 (2004).
  - [18] 濱口明宏, 神原誠之, 横矢直和: 装着した 3 次元位置姿勢センサによるウェアラブルユーザの自己位置推定, 電子情報通信学会技術研究報告 (MVE), Vol.105, No.566, pp.19-24 (2006).
  - [19] 羽原寿和, 町田貴史, 小川剛史, 竹村治雄: 画像マーカを用いた屋内位置検出機構とその評価, 電子情報通信学会技術研究報告 (MVE), Vol.102, No.220, pp.65-70 (2002).
  - [20] 石塚宏紀, 上坂大輔, 黒川茂莉, 渡邊孝文, 村松茂樹, 小野智弘: BLE シグナルと PDR によるハイブリッド屋内測位手法の基礎検討—Open Beacon Field Trial 参加における実験結果の共有, 信学技報, Vol.114, No.31, MoNA2014-10, pp.133-138 (2014).
  - [21] 田村侑介, 永長知孝: Bluetooth を用いた無線 LAN ポジショニングシステムの性能改善に関する検討, 電子情報通信学会技術研究報告, ITS, Vol.110, No.469, pp.23-28 (2011).
  - [22] Dinesh, M. and Hideyuki, T.: Opening Up Indoors: Japan's Indoor Messaging System, IMES, *GPS World*, Vol.22, No.5, pp.38-46 (2011).
  - [23] 石川尊之, 堀 鉄郎, 河崎晋也, 相澤清晴: ライフログに向けた加速度センサからの歩行検出, 電子情報通信学会総合大会講演論文集, Vol.2004, p.314 (Mar. 2004).
  - [24] 伊藤恵美子, 大口 敬, 鹿田成則: 信号交差点における歩行者信号切り替わり時と歩行者横断挙動に関する解析, 土木計画学研究・講演集 (神戸大), 企画論文部門 (2004).
  - [25] 森 傑, 奥 俊信: 自由散策行動にみられるアクションの特性—都市空間におけるアクトファインディングに関する基礎的研究—, 都市計画論文集「学術研究論文発表会論文」, Vol.37, pp.31-36 (2002).



寺田 努 (正会員)

1997 年大阪大学工学部情報システム工学科卒業。1999 年同大学院工学研究科博士前期課程修了。2000 年同大学院工学研究科博士後期課程退学。同年より大阪大学サイバーメディアセンター助手。2005 年より同講師。2007 年神戸大学大学院工学研究科准教授, 現在に至る。2004 年より特定非営利活動法人ウェアラブルコンピュータ研究開発機構理事, 2005 年には同機構事務局長を兼務。2004 年には英国ランカスター大学客員研究員, 2010 年より科学技術振興機構さきがけ研究員を兼務。博士 (工学)。アクティブデータベース, ウェアラブルコンピューティング, ユビキタスコンピューティングの研究に従事。IEEE, ACM, 電子情報通信学会, 日本データベース学会, ヒューマンインタフェース学会, 芸術科学会の各会員。



宮下 浩一 (学生会員)

2006 年大阪大学工学部電子情報エネルギー工学科卒業。2008 年同大学院情報科学研究科マルチメディア工学専攻修士前期課程修了。現在, 株式会社三菱総合研究所に在籍し, ITS に関する調査コンサルティング業務に従事。

また, 2014 年より神戸大学大学院工学研究科電気電子工学専攻博士後期課程に在籍。技術士 (情報工学部門)。