

# Wi-Fi位置計測のための計測エラー検出および修正手法

津田 祐輝<sup>1,a)</sup> 孔 全<sup>1,b)</sup> 前川 卓也<sup>1,c)</sup>

受付日 2013年4月8日, 採録日 2013年10月9日

**概要:** 本論文では, Wi-Fi 位置計測における大きい誤差を持つ計測エラーの検出, 修正を行う手法を提案する. Wi-Fi 位置計測において, 参照するアクセスポイントの移動や位置データベースの劣化により, 数十 m を超える計測誤差が発生することがある. 提案手法では, 過去の Wi-Fi 計測位置の履歴から現在位置を推定し, Wi-Fi 計測位置と比較することで Wi-Fi 計測エラーを検出, 修正する. このとき, ユーザのコンテキストによって, 位置推定の性能が大きく変化するため, 様々な特性を持つ位置推定手法を複数用いることで, コンテキストアウェアなエラー検出フレームワークを構築する. 評価実験では, 実際に様々なコンテキストを想定して, 収集した Wi-Fi 位置計測データを使用し, 提案手法の有効性を検証した.

キーワード: Wi-Fi 位置計測, 計測エラー, 教師あり機械学習

## Positioning Error Detection and Correction Methods for Wi-Fi Based Localization

YUKI TSUDA<sup>1,a)</sup> QUAN KONG<sup>1,b)</sup> TAKUYA MAEKAWA<sup>1,c)</sup>

Received: April 8, 2013, Accepted: October 9, 2013

**Abstract:** In this paper, we propose methods that detect and correct large errors in Wi-Fi based localization. Because of the movement of reference Wi-Fi access points, we sometimes get a Wi-Fi position that has more than dozens of meters errors. In this paper, we detect and correct such errors automatically by comparing current Wi-Fi position with current positions predicted from a history of Wi-Fi positions. However, performances of the position predictors change greatly with the user's context (e.g. migration speed and density of access points around the user). So, we design context-aware frameworks for error detection and correction by employing an ensemble of predictors that have strengths and weakness for different contexts.

**Keywords:** Wi-Fi based localization, Positioning error, Supervised machine learning

### 1. はじめに

携帯端末を用いた位置計測技術の発展にともなって, 位置情報を用いた様々なサービスが急速に普及している. スマートフォンによるナビゲーションはもとより, SNS などでの位置情報の共有や位置情報ライフログへの応用も進みつつある. 位置計測技術としては, GPS (Global Positioning System) が長年利用されている. GPS 以外の

屋外位置計測技術としては, Wi-Fi の無線 LAN アクセスポイント (以下, AP) からの電測情報を用いて位置を計測する Wi-Fi 位置計測が注目されている [1], [2], [3]. Wi-Fi 位置計測では, 携帯端末が受信した Wi-Fi の電波強度とその MAC アドレスの情報 (Wi-Fi スキャン) を用いて, Wi-Fi 位置情報サービスのプロバイダへ問合せを行うことで, プロバイダにより推定された位置情報を得る. Wi-Fi 位置計測手法は, 主に訓練段階と計測段階からなる. 訓練段階では, あらかじめ様々な場所において Wi-Fi スキャンとその場所の緯度経度情報を収集して, データベースに格納しておく. 計測段階では, ユーザが緯度経度が未知の場所において計測した Wi-Fi スキャンを用いて, そのユーザの緯度経度を推定する. 具体的には, そのスキャンとデータベー

<sup>1</sup> 大阪大学大学院情報科学研究科  
Graduate School of Information Science and Technology,  
Osaka University, Suita, Osaka 565-0871, Japan

a) tsuda.yuuki@ist.osaka-u.ac.jp

b) kong.quan@ist.osaka-u.ac.jp

c) maekawa@ist.osaka-u.ac.jp

スに格納されたそれぞれのスキャンとの距離をユークリッド距離などを用いて計算し、最も近いスキャンに紐づく緯度経度を、ユーザの緯度経度として出力する。

この2手法にはそれぞれ長所と短所が存在する。GPSは、GPS電波の届かない屋内や地下では位置計測を行えない。また、多くの電力を消費するため、電池容量に制限のある携帯端末においては、長時間の利用ができないという問題がある。しかし、周囲が開けた場所であれば、その誤差は数m程度ときわめて精度の高い位置計測が可能である。一方、Wi-Fi位置計測は、APが周囲に存在しない場所や利用するWi-Fi位置情報サービスがカバーしていない範囲では位置計測が行えないが、周囲にAPが存在すれば、屋内においても位置計測を行うことができる。また、その消費電力はGPSと比べて少なく、携帯端末での長時間の使用に適している。しかし、その計測誤差は数十m程度とGPSに比べると大きく、計測精度は劣っている。GPSとWi-Fi位置計測における、このような特徴の違いから、多くの研究者によりGPSとWi-Fi位置計測のハイブリッド計測手法が提案されている。これらの手法では、それぞれの長所である高い計測精度や少消費電力を同時に達成することを目指している。また、それぞれの位置計測のカバーエリアを補い合うことが可能である。

一方、Wi-Fi位置計測においては、数百mから数km、数十km以上というきわめて大きな計測誤差が発生することがある。このような計測誤差を持つ位置計測（以下、計測エラー）は、Wi-Fi位置計測データベースの劣化やモバイルWi-Fiルータの移動によって引き起こされる[4], [5], [6]。Wi-Fi位置情報サービスへ登録されているAPの位置と実際のAPの位置が異なる場合、そのAPを観測した端末の位置は、そのAPが登録されている位置として誤って計測される（このようなAPを異常APと呼ぶ）。そこで、Wi-Fi位置計測における大きな計測エラーの自動検出と計測位置の自動修正を目指す[7]。本研究では、計測エラーの検出を外れ値検出と見なす。過去のWi-Fi計測位置やその過程で得られたデータから現在位置を推定する「位置推定器」を用意し、その推定位置と現在のWi-Fi計測位置を比較することにより、外れ値を検出する。しかし、携帯端末の計測を行った環境（APの密度や過去の位置計測の密度など）が位置推定器の性能に多大な影響を与えるため、単一の位置推定器により正確に現在位置を推定することは困難である。たとえばトンネルを通り抜けた後などの、直前のWi-Fi位置観測数が少ない状況ではカルマンフィルタなどの過去の履歴を用いる手法の精度は低くなる。また、観測されたWi-Fi信号強度情報を用いて他の位置情報プロバイダに問合せを行うような位置推定器を考えたとき、周囲のAPの数が少ない場所ではその手法の精度は低くなる。そこで、このような様々な特徴を持つ位置推定器を複数用意し、さらに計測環境を考慮することで頑健な計測エラー

の検出を行う。計測エラーの検出を実現することで、計測エラーが検出された場合は、信頼性の高いGPSへと切り替えるハイブリッド位置計測が可能となる。また計測エラーの修正では、それぞれの位置推定器の出力を、その計測環境に応じた信頼性を考慮したうえで組み合わせることで、より正確な位置推定を実現する。計測エラーの修正を実現することで、ユーザ端末が屋内や地下にある場合や端末のバッテリー残量が少ない場合といったGPSに切り替えられない状況において、より正確な位置を提供できる。

## 2. Wi-Fi計測エラーの検出および修正手法

### 2.1 概要

本研究では、携帯端末を持ったユーザがライフログやナビゲーションなどのアプリケーションにおいて、定期的にWi-Fi位置情報を計測している状況を前提とする。このとき、時刻 $1 \sim T$ のWi-Fi位置計測の現在位置 $(x_1, y_1), \dots, (x_T, y_T)$ 、および、時刻 $1 \sim T$ のWi-Fiスキャン $WS_1, \dots, WS_T$ のデータが端末によって計測、保持される。携帯端末が時刻 $T$ にWi-Fi位置計測を実行した際に、計測位置 $(x_T, y_T)$ と問合せに用いたWi-Fiスキャン $WS_T$ の2種類のデータが得られる。Wi-Fiスキャン $WS_T$ は、携帯端末が観測した周囲のAPのMACアドレスと受信信号強度からなるデータである。このようなデータを用いて、本研究では $(x_T, y_T)$ に対する「計測エラーの検出」と「計測エラーの修正」を、教師あり機械学習のアプローチにより実現する。計測エラーの検出に対応する手法の出力は、対象とする時刻における計測エラーの有無であり、計測エラーの修正に対応する手法の出力は、対象とする時刻における修正された現在位置座標である。また、上記の2手法は消費電力の面からサーバコンピュータ上で実行されることを前提としている。両手法は教師あり学習を用いるため教師データとして、時刻 $t$ の正確な現在位置座標 $(X_t, Y_t)$ が必要となる。本研究では、正確な現在位置座標としてGPSによる位置座標とESM (Experience sampling method) によって得られた位置座標を使用する。Wi-Fi位置計測時に屋内にいるなどの理由でGPSによる位置計測が行えなかった場合に、実験参加者へ現在位置座標の入力を求めることで、正確な位置を欠かさず取得する。

### 2.2 計測エラーの検出

計測エラーの検出においては、2クラス分類器を使用して「計測エラーあり」もしくは「計測エラーなし」の2クラスに分類するモデルを教師あり機械学習により学習する。以降で説明する実験においては、決定木を分類器として用いた。モデルの入力としては、図1のように特徴抽出を行って構築した特徴ベクトルを使用する。この特徴抽出では、大きく分けて以下の2種類の特徴量を抽出し、それらをすべて連結することで特徴ベクトルを構築する。このと

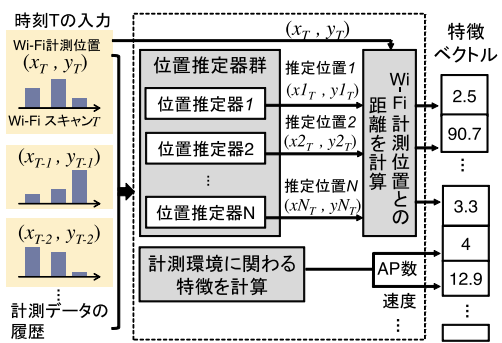


図 1 計測エラーの検出に用いる特徴ベクトルの作成  
 Fig. 1 Constructing feature vector for error detection.

き、計算された以下の特徴の各要素間の正規化は行っていない。これは、以降で分類器として用いる決定木がベクトルの要素ごとに分類の判定を行うためである。

**位置推定器による推定位置と Wi-Fi 計測位置との距離：**位置推定器  $n$  による推定位置  $(x_{nT}, y_{nT})$  と Wi-Fi 計測位置  $(x_T, y_T)$  とのユークリッド距離を特徴量として使用する。図 1 に示すように、位置推定器  $n$  は前述の入力データから位置座標  $(x_{nT}, y_{nT})$  を推定する。その後、位置推定器による推定位置  $(x_{nT}, y_{nT})$  と Wi-Fi 計測位置  $(x_T, y_T)$  間の距離を算出する。この距離が小さければ、位置推定器による予測と観測が近く、計測エラー発生の可能性が低い。  
**計測環境の特徴：**計測環境の特徴は、問合せに用いた AP の数や携帯端末の移動速度といった、Wi-Fi 位置計測の性能や各位置推定器の性能に影響を与える特徴量である。

2.2.1 位置推定器の詳細

本研究では、計測環境に応じた計測エラーの検出を行うために、以下の様々な特徴を持つ位置推定器を使用する。  
**カルマンフィルタ：**線形的に表現できる移動軌跡をトラッキングする用途で利用されている技術であるカルマンフィルタ [8] を使用して位置推定を行う。そのアルゴリズムは、予測と更新の 2 つのステップからなる。まず任意の時刻  $t$  のトラッキング対象の状態を示すベクトルを  $p_t$  とする。この場合は、緯度と経度の 2 次元ベクトルとなる。すると、時刻  $t$  の対象の状態は  $\hat{p}_{t+1}^- = A\hat{p}_t$  のように予測される (事前予測)。  $A$  は遷移行列であり、今回は等速直線運動を仮定したものを用いている。すなわち、 $\hat{p}_t$  と  $\hat{p}_{t-1}$  から求めた速度をそのまま利用している。ただし、 $\hat{p}_t$  は、更新の際に実際の観測を用いて修正された時刻  $t$  の事後予測であり、 $\hat{p}_t = \hat{p}_t^- + K_t(m_t - H\hat{p}_t^-)$  のように表される。  $m_t$  は任意の時刻  $t$  の観測 (緯度経度) であり、 $H\hat{p}_t^-$  は事前予測の予測観測値である。今回は  $H$  は単位行列としている。すなわち、右辺第 2 項は観測と事前予測との誤差にカルマンゲイン  $K_t$  を掛けたものとなる。カルマンゲインは、 $\hat{p}_t^-$  と  $m_t$  のどちらに重みをおいて  $\hat{p}_t$  を推定するかを調整する。カルマンフィルタのさらなる詳細に関しては文献 [8] を参照されたい。この位置推定器では  $\hat{p}_t^-$  に含まれる緯度経度を

時刻  $t$  における推定位置とする。

この位置推定器は、Wi-Fi 計測位置の過去の履歴が十分に存在し、また移動速度の変化が小さい場合に、良い性能を得る。しかし、Wi-Fi 位置計測の履歴が少ない場合は、過去の軌跡の情報が正確に得られずに位置推定の精度が低下してしまう。また過去の履歴が最低 2 つ以上存在しなければ、この位置推定器は位置推定を行うことができない。この場合、対応する特徴ベクトルの要素を欠損値として扱う。  
**改良カルマンフィルタ：**前述のカルマンフィルタを改良したフィルタを用いて位置推定を行う。本研究が対象とする入力データは、計測エラーによるきわめて大きな誤差を含むことがある。カルマンフィルタは、予測と更新のステップを入力データごとに繰り返すため、1 つの大きな誤差がその後の予測に重大な影響を及ぼしてしまう。そこで、本研究では任意の時刻  $t$  の Wi-Fi 計測位置  $m_t$  と時刻  $t-1$  の更新位置  $\hat{p}_{t-1}$  から求めた速度が閾値を超えた場合は、カルマンフィルタの更新式における  $K$  を零行列として更新を行う。すなわち、計測位置  $m_t$  を用いて更新を行わない。  
**パーティクルフィルタ：**携帯端末の利用者による実際の移動には、車や電車を乗降するといった移動手段の変更や交通信号機などによる移動速度の急激な変化が起こることが考えられる。そのような場合に対応するため、非線形な状態遷移を行うシステムの状態を予測するパーティクルフィルタ [9] を使用する。そのアルゴリズムは、サンプリング、重み計算、リサンプリングの 3 ステップからなる。サンプリングにおいては、時刻  $t$  のパーティクル  $p_t$  から、新たなパーティクルを作成し、移動モデル  $p_{t+1}^i = Ap_t + w$  に基づいてそのパーティクルを移動させる。このパーティクルがトラッキング対象の状態 (位置) を表す。上記のカルマンフィルタの実装との違いは、移動に平均 0 のガウシアンノイズ  $w$  を含むことと、1 つの  $p_t$  からガウシアンノイズに従って複数の  $p_{t+1}^i$  をサンプリングすることである ( $i$  はパーティクルの識別子)。今回の実装では 1 つのパーティクルを 5 つに分裂させている。また移動速度の急激な変化に対応するため、 $w$  の標準偏差を大きく設定している (時刻  $t-1$  から  $t$  に進んだ距離)。重み計算においては、観測を用いてパーティクルへの重みづけを行う。本研究では、Wi-Fi 計測位置  $m_{t+1} = (x_{t+1}, y_{t+1})$  に近いパーティクルは、実際にその場に携帯端末が位置する可能性が高いと考えられるため、近いパーティクルに大きい重みを与える。 $p_{t+1}^i$  の重みは、 $m_{t+1}$  を平均とする正規分布の確率密度関数  $w_{t+1}^i = \mathcal{N}(p_{t+1}^i | m_{t+1})$  により計算する。リサンプリングにおいては、重みが小さいパーティクルの排除を行う。今回の実装では、小さい順に  $\frac{4}{5}$  のパーティクルを削除する。この 3 ステップを繰り返すことで軌跡の推定を行う。この位置推定器の出力としては、サンプリングのステップで得られたパーティクルの平均位置を使用する。

この位置推定器は、Wi-Fi 計測位置の過去の履歴が十分に



に存在する場合に良い性能を得る。ただし、パーティクルの速度を求めるために直前の履歴が2つ必要であるため、過去の履歴が最低2つ以上存在しなければ、この位置推定器は位置推定を行うことができない。この場合、対応する特徴ベクトルの要素を欠損値として扱う。

**他プロバイダへの問合せ：**この位置推定器は、時刻  $T$  の Wi-Fi スキャン  $WS_T$  を用いて、他の Wi-Fi 位置情報サービスのプロバイダへ問い合わせることにより、推定位置を得る。この位置推定器は、Wi-Fi 計測位置の過去の履歴が存在しない場合でも位置推定が可能であるが、 $WS_T$  に含まれる AP の数が少ない場合は正確な位置推定が行えない場合がある。さらに Wi-Fi 位置計測を行うため、異常 AP の影響を受けることが考えられる。しかし、Wi-Fi 位置情報サービスのプロバイダによって、AP の登録方法や登録状況が異なるため、各プロバイダのカバーする範囲や位置推定の精度に違いが生じる。そのため、他のプロバイダの問合せから得られた推定位置により、対象とするプロバイダの Wi-Fi 計測エラーの検出を行うことができる。このとき、他のプロバイダに問い合わせることによる位置推定は、携帯端末上で行われる Wi-Fi 位置計測と本質的には同じものである。すなわち、携帯端末上の Wi-Fi 位置計測と同様に携帯端末上で動作させることも考えられる。今回の実装では1つのプロバイダのみにしか問合せを行っていないが、多数のプロバイダを利用できる場合、この推定器も多数用意されることが考えられる。このとき、それらの推定器を Wi-Fi 位置計測と同様に携帯端末上で動かすことは、バッテリーリソースの限られる携帯端末にとって現実的ではない。そのため、このほかのプロバイダ問合せを行う位置推定器を、携帯端末上のネイティブな Wi-Fi 位置計測とは異なり、サーバ上で実行される他の推定器と同様に扱うこととする。

**修正 Wi-Fi スキャンによる問合せ：**この位置推定器は、異常 AP を Wi-Fi スキャン  $WS_T$  から取り除き、修正後の Wi-Fi スキャン  $WS'_T$  を用いて、入力データの Wi-Fi 位置計測に用いたプロバイダへ再度問合せを行うことで位置推定を行う。この手法は、異常 AP はその他の AP に比べてその数が少ないという前提に基づいて動作する。まず、スキャンに含まれる AP をいくつかのグループに分割する。異常 AP の数が（グループの数の半数より）少ない場合、過半数のグループには異常 AP が含まれないことになる。そのようなグループを発見することにより位置の推定を行う。このとき、1つのグループに含まれる AP 数は最低2つであるとした。これは、単一の AP のみを用いて行った位置計測結果の精度は高くないと考えられるためである。また、今回実装に用いた位置計測プロバイダ (Google Geo Location) は、単一の AP のみからなる問合せを受け付けない。 $WS_T$  に含まれる AP 数が  $2g$  より少ない場合は、それぞれのグループに含まれる AP をその要素数が2になる

までランダムに追加する。以下に、具体的な例を用いて、この位置推定器が行う位置推定の4つのステップを説明する。ここでは、説明のために Wi-Fi スキャン  $WS_T$  が6つの AP (AP1, ..., AP6) からの信号を含み、そのうち1つが異常 AP (AP1) であるとする。

1. Wi-Fi スキャン  $WS_T$  に含まれる AP を  $g$  個のグループにランダムに分割する。このとき、それぞれのグループ内の AP の数は、 $\left\lceil \frac{\# \text{APs in } WS_T}{g} \right\rceil$  を超えないものとする。また、 $g$  は奇数である必要があり、ここでは  $g = 3$  として説明を行う。例においては、グループ1に AP1 と AP2、グループ2に AP3 と AP4、グループ3に AP5 と AP6 というように分割される。

2. それぞれのグループ内の AP の信号強度情報から構成される Wi-Fi スキャンを作成し、Wi-Fi 位置情報サービスに問合せを行う。そして、 $g$  個の Wi-Fi 計測位置を得る。この例においては、3つのグループに対応する Wi-Fi スキャンが作成され、Wi-Fi 計測位置  $(x_{g1T}, y_{g1T})$ ,  $(x_{g2T}, y_{g2T})$ ,  $(x_{g3T}, y_{g3T})$  を得る。

3. 得られた Wi-Fi 計測位置を比較し、異常 AP を含むグループを特定する。例において、グループ1は異常 AP を含むため、対応する Wi-Fi 計測位置  $(x_{g1T}, y_{g1T})$  は、大きな誤差を持つはずである。そのため、Wi-Fi 計測位置  $(x_{g1T}, y_{g1T})$  は、他の2つの Wi-Fi 計測位置から遠い位置に存在すると考えられる。そこで、離れた計測位置を自動的に特定するために、得られた  $g$  個の Wi-Fi 計測位置に対して階層的クラスタリングを行う。そして、最も大きいクラスタ内の要素の数が、 $\left\lfloor \frac{g}{2} \right\rfloor$  を超えたとき、クラスタリングを終了する。得られた最大のクラスタは、異常 AP の影響を受けた Wi-Fi 計測位置を含まず、携帯端末の実際の位置に近い Wi-Fi 計測位置のみで構成されることが期待される。例においては、グループ2とグループ3に対応する Wi-Fi 計測位置  $(x_{g2T}, y_{g2T})$ ,  $(x_{g3T}, y_{g3T})$  が最大のクラスタを構成することとなる。ここで、終了条件が満たされなかった場合、最終的に2つのクラスタが残ることとなる。 $g$  が偶数であった場合、多数決ではどちらのクラスタを異常 AP を含まないクラスタとするか判定できなくことがあるため、 $g$  を奇数としている。

4. 最大のクラスタ内の Wi-Fi 計測位置を得る問合せに用いた AP すべてを含む Wi-Fi スキャン  $WS'_T$  を作成する。そして、Wi-Fi 位置情報サービスへ問合せを行い、推定位置を得る。この例では AP3, AP4, AP5, AP6 を用いて、Wi-Fi スキャン  $WS'_T$  を作成する。そして、問合せを行うことで異常 AP (AP1) の影響を受けていない Wi-Fi 計測位置を得る。

## 2.2.2 計測環境の特徴の詳細

本研究では、携帯端末の計測環境の情報をとらえるため、以下の特徴量を使用する。

**計測間隔：**この特徴は時刻  $T$  と前回の計測時刻  $T-1$  の時

間隔に対応し、 $time(T) - time(T - 1)$  で表される。ただし  $time(T)$  は時刻  $T$  のタイムスタンプであり単位は分である。トンネル内を移動する場合など、周囲に AP が存在しない状況では、長時間 Wi-Fi 位置計測が行えずに計測間隔は大きくなり、軌跡を用いる位置推定器の性能が下がってしまう。この特徴量を使用することで、そのような性能の低下を考慮することが可能となる。

**移動速度と移動距離：**時刻  $T$  と  $T - 1$  において計測された位置  $(x_{T-1}, y_{T-1})$ ,  $(x_T, y_T)$  から算出され、それぞれ  $\sqrt{(x_T - x_{T-1})^2 + (y_T - y_{T-1})^2}$  と  $\frac{\sqrt{(x_T - x_{T-1})^2 + (y_T - y_{T-1})^2}}{time(T) - time(T-1)}$  で表される。ただし、単位はメートルである。この2つの値が極端に大きければ、時刻  $T$  において計測エラーが発生していることが予想される。

**移動速度の分散と平均：**平均  $\bar{v}$  は  $\frac{\sum_{t=T-n}^T v_t}{n+1}$  のように表される。ただし、 $v_t$  は時刻  $t$  の速度である。また、分散は  $\frac{\sum_{t=T-n}^T (\bar{v} - v_t)^2}{n+1}$  のように表される。携帯端末の移動速度が変化した場合、分散値は大きくなり等速運動を前提とするカルマンフィルタを用いた位置推定器の性能は低下する。

**AP 数：**時刻  $T$  における Wi-Fi スキャン  $WS_T$  内に含まれる AP の数  $Num(WS_T)$  を特徴量とする。AP 数が少ない場合は、Wi-Fi 位置計測自体の精度が低くなるため、再度 Wi-Fi 位置計測を使用する位置推定器の性能は低下する。

**受信信号強度の平均と分散：** $WS_T$  内に含まれる  $i$  番目の AP の受信信号強度を  $RS_i$  とすると、平均と分散は  $\frac{\sum_i RS_i}{Num(WS_T)}$  と  $\frac{\sum_i (RS_i - \bar{RS})^2}{Num(WS_T)}$  で表される。ただし  $\bar{RS}$  は平均である。これらの特徴は周囲の電波環境に影響されるため、Wi-Fi 位置計測を使用する位置推定器の性能に影響を与える可能性がある。

**モバイル Wi-Fi ルータに関する特徴：**携帯端末の周囲に存在するモバイル Wi-Fi ルータは、Wi-Fi 位置計測に悪影響を及ぼす。たとえば、モバイル Wi-Fi ルータを所持している人物が、同じ車両に乗り込んでいる場合、携帯端末は移動しているにもかかわらず、Wi-Fi 位置計測の結果はそのモバイル Wi-Fi ルータがプロバイダに登録されている場所を指し続けることがある。このような場合、携帯端末はモバイル Wi-Fi ルータからの信号を受信し続けることになる。しかし、家庭やオフィスに固定設置された AP に関しては、携帯端末が高速で移動し続けるため、連続的に観測されることは少ない。すなわち、時刻  $T - 1$  に観測された固定設置 AP は、時刻  $T$  においては観測されることは少ない。そこで、モバイル Wi-Fi ルータに関する特徴として、時刻  $T - 1$  と  $T$  のいずれかで観測された AP の数  $Num(WS_T \cup WS_{T-1})$  に対して、時刻  $T - 1$  と  $T$  の両方で観測された AP の数  $Num(WS_T \cap WS_{T-1})$  の割合を算出して使用する。すなわち、 $\frac{Num(WS_T \cap WS_{T-1})}{Num(WS_T \cup WS_{T-1})}$  で表される。さらに、過去  $n$  分間の Wi-Fi スキャン  $WS_{T-n}, \dots, WS_T$  を基に算出する特徴も使用する。まず、過去  $n$  分間に 1 度でも観測された AP

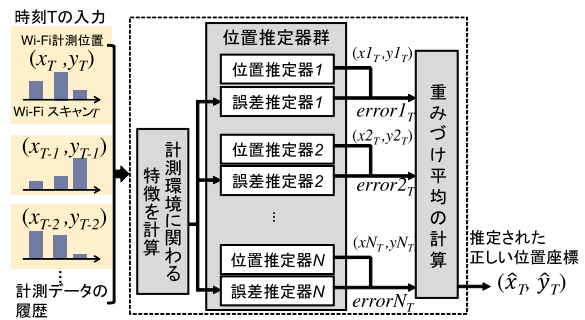


図 2 計測エラーの修正手法の概要

Fig. 2 Overview of correct coordinate prediction.

に関して、過去  $n$  分間の Wi-Fi スキャン  $WS_{T-n}, \dots, WS_T$  で観測された回数  $O_i = Num(n, AP_i)$  を数える。そして、その出現回数の分散  $\frac{1}{|AP_s|} \sum_{AP_i \in AP_s} (\bar{O} - O_i)^2$  を特徴量として用いる。ただし、 $AP_s$  は過去  $n$  分間に 1 度でも観測された AP の集合を、 $\bar{O}$  は  $O_i$  の平均を示す。このとき、モバイル Wi-Fi ルータによる AP の観測回数は大きな値をとるが、固定設置 AP の観測回数は 1, 2 回となる。すなわち、モバイル Wi-Fi ルータの所持者が、同じバスや車両に乗り込んでいる場合、その出現回数の分散は増加する。この2つの特徴量を用いることで、モバイル Wi-Fi ルータの存在をある程度、検知することができる。

本研究では、過去のどれだけのデータを参照して特徴計算するかを決める  $n$  の値は 5 分とする。

### 2.3 Wi-Fi 位置計測における計測エラーの修正

図 2 は、計測エラーの修正手法の全体像を示す。ここでは、上記の手法により算出された位置推定器の推定位置と計測環境の特徴量を再利用する。前述のとおり、各位置推定器の性能は計測環境に応じて変化するため、各推定位置を一律に考慮して、計測エラーの修正を行うべきではない。そこで、計測環境の特徴量から時刻  $T$  の計測環境における各推定位置の誤差を推定して、その値を考慮することで、計測エラーの修正を行う。各推定位置の誤差は、図 2 の誤差推定器において時刻  $T$  の計測環境の特徴量から、回帰分析を用いて、時刻  $T$  における各推定位置の誤差  $e_{nT}$  を推定する。このとき、位置推定器の誤差とは GPS や ESM によって得られた実際の携帯端末の位置と位置推定器の推定位置間の距離である。このようにして推定された推定位置の誤差  $e_{nT}$  を用いて、式 (1) に示す重み付け平均位置  $(\hat{x}_T, \hat{y}_T)$  の算出を行い、修正された計測位置として出力する。

$$(\hat{x}_T, \hat{y}_T) = \frac{\sum_{i=n}^N w_n \cdot 1/e_{nT}(x_{nT}, y_{nT})}{\sum_{n=1}^N w_n \cdot 1/e_{nT}} \quad (1)$$

式 (1) では、推定された誤差  $e_{nT}$  の逆数を、推定位置  $(x_{nT}, y_{nT})$  の信頼性を表す係数として用いている。推定誤差  $e_{nT}$  の逆数を係数として使用することで、携帯端末の実際の位置に近いと推定された推定位置をより重く考慮でき

る。また、 $w_n$  はトレーニングデータから得られた  $n$  番目の推定器の重みであり、認識精度（適合率）に対応する。適合率の算出方法については 4.1 節を参照されたい。

### 3. 評価実験

#### 3.1 実験データ

実験対象データの収集のため、複数人の参加者にデータ収集用の携帯端末 (Google Nexus One) を持ち歩いてもらった。そして、携帯端末上のデータ収集用のアプリケーションを使用して、データの収集を行った。Wi-Fi 位置情報サービスのプロバイダは、Google Location Service [10] を使用し、1 分ごとに位置計測を行った。Google Location Service では、セルラ基地局から得られる電波情報も補足的に計測に用いている。本研究の有効性を評価するためには、様々な計測環境に対応したデータを用意する必要がある。そこで、以下の 3 種類の条件において、移動データを収集した。収集したそれぞれの軌跡データは、始点と終点がすべて異なるものである。

- **通勤通学データセット**：このデータセットは、参加者の通勤通学時にデータ収集用の携帯端末を持ち歩いてもらうことで収集した。移動方法は電車とバス、徒歩の組合せであり、速い速度での移動時や移動速度の急激な変化を想定している。
- **ビル街データセット**：参加者がデータ収集用の携帯端末を持って、ビル街を徒歩で散歩することにより、移動データを収集した。参加者は、ビル間の歩道や商業ビル、駅建物内を歩く。また、ショッピングなどにより大きな移動が発生しない状況も想定している。
- **地下街データセット**：参加者がデータ収集用の携帯端末を持って、地下街を徒歩で移動することにより、移動データを収集した。このデータセットは、周囲の AP が少数であるような環境を想定している。

表 1 は、これらのデータセットの概要を示している。また、図 3 は Wi-Fi 計測位置の誤差の分布を、図 4 はその誤差の CDF (累積分布関数) を示している。ここで、誤差とは Wi-Fi 計測位置と GPS や ESM による計測位置間の距離を表している。図 3 より、0~150m には多数の誤差

表 1 各データセットの概要  
Table 1 Overview of our data set.

	通勤通学	ビル街	地下街	全体
軌跡数	11	10	10	31
平均計測時間 (分)	84.8	31.6	26.7	48.9
総特徴ベクトル数	317	234	172	723
位置計測数 /1 時間	24.8	44.8	39.1	31.5
エラー率 (> 150m)	32.8%	9.8%	20.3%	22.4%

が見られるが、150m 以上では誤差の発生がまばらである。しかし、図 4 から数十万 m 程度の誤差も発生しており、広い範囲で誤差が発生していることが分かる。ただし、この計測誤差は GPS の計測結果との比較であり、GPS の計測結果にも約 5m 程度の計測誤差が含まれることに注意されたい [11]。以上から、150m 以上の誤差を持つ Wi-Fi 計測位置を計測エラーとし、以降の評価を行う。また、iPhone における Wi-Fi 位置計測の誤差を検証した文献 [11] では、誤差 100m 以内の観測が約 63% 程度となっており、Wi-Fi 位置計測の最大誤差を 150m とするのは妥当と考える。ただし、提案手法は任意に設定したエラーの検出も可能であり、以降ではエラーの値を変化させたときの評価も行う。

図 5 に、データセットにおける移動速度とそのときのエラー率との関係を示す。移動速度が小さい場合は極端にエラー率が小さく、その場合は高い正解率でエラーの有無が推定できるだろう。さらに図 6 に、データセットにおける端末の周辺の AP 数とそのときのエラー率との関係を示

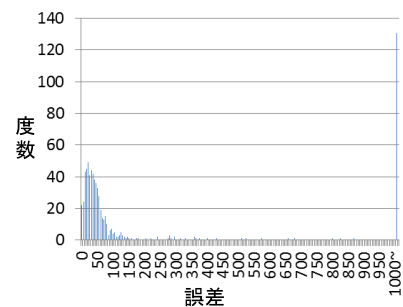


図 3 Wi-Fi 計測位置の誤差の分布

Fig. 3 Distribution of Wi-Fi positioning errors.

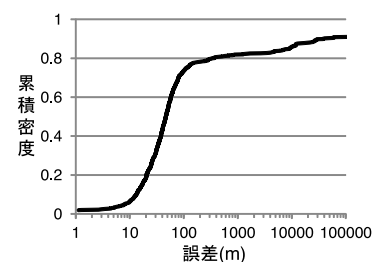


図 4 Wi-Fi 計測位置の誤差の CDF

Fig. 4 CDF of Wi-Fi positioning errors.

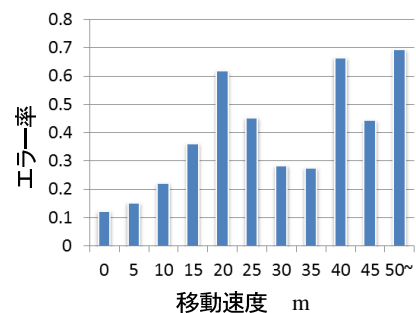


図 5 移動速度と計測エラー率の関係

Fig. 5 Relationship between speed and error rate.



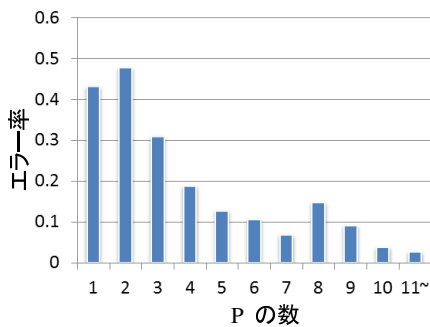


図 6 スキャン内の AP とエラー率の関係

Fig. 6 Relationship between # APs and error rate.

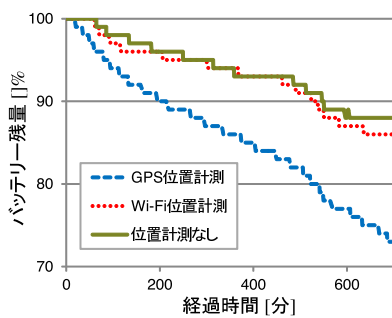


図 7 バッテリー残量の推移

Fig. 7 Transitions of battery levels.

す。AP 数が少ないときは推定に用いることができる情報が少ないため、推定精度が低い。一方、図 7 は 1 分ごとに GPS および Wi-Fi により位置計測を行った際の携帯端末の電池残量の推移を示している（提案手法の電力消費は原理的に Wi-Fi 位置計測と等価である）。さらに、位置の計測をいっさい行わないときの電池残量の推移も示している。この結果から、待機時にも多くの電力を消費していることが分かるが、GPS を用いることでさらに消費電力が増大していることが分かる。一方、Wi-Fi 位置計測の消費電力は小さく、待機時の消費電力と大きく変わらないことが分かる。

## 4. 実験結果

### 4.1 Wi-Fi 位置計測における計測エラーの検出

評価手法：評価のために、表 1 の実験データに対して、leave-one-out 交差検定を行った。すなわち、全 31 個のデータのうち 1 つのデータをテストデータとし、残る 30 個のデータをモデルの学習データとして用いることを繰り返すことで、すべての移動データに対して検定を行った。計測エラーの検出を行う 2 クラス分類器としては、C4.5 決定木を用いた。提案手法の有効性を評価するための比較手法として、前述の位置推定器それぞれによる推定位置  $(x_{nT}, y_{nT})$  と Wi-Fi 計測位置  $(x_T, y_T)$  間の距離のみを特徴量として構築した 2 クラス分類器 (C4.5 決定木) を用いる手法を用意した。評価項目としては、各時刻における「計測エラーあり」「計測エラーなし」という 2 クラスへの分類

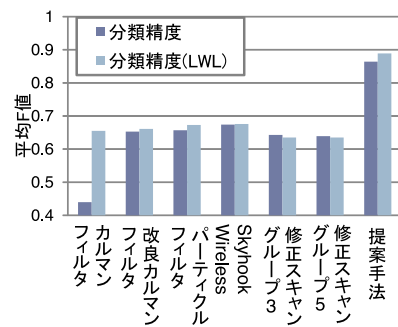


図 8 各手法の分類精度

Fig. 8 Accuracies of naive and proposed methods.

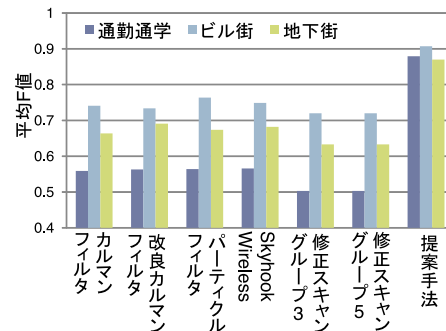


図 9 データセットの各手法の分類精度

Fig. 9 Accuracies of naive and proposed methods for each data set.

結果の適合率、再現率、F 値および、2 クラスの F 値の平均である平均 F 値を使用する。ただし、適合率はあるクラスに正しく分類された特徴ベクトルの数と、そのクラスに分類された特徴ベクトルの数との比である。再現率は、あるクラスに正しく分類された特徴ベクトルの数と、実際にそのクラスに属する全特徴ベクトルの数との比である。また、F 値は  $\frac{2 \times \text{適合率} \times \text{再現率}}{\text{適合率} + \text{再現率}}$  で表される。また、他のプロバイダへの問合せを用いる位置推定器においては、“Skyhook wireless” [12] への問合せにより位置推定を行った。

評価結果・考察：図 8 に各手法による計測エラー検出の平均 F 値を示す。表 1 にあるように「計測エラーなし」のサンプル数に対して、「計測エラーあり」のサンプル数が少ないため、「計測エラーあり」のサンプルに重みづけ学習 (LWL: Locally Weighted Learning) を行った結果も図 8 に示している。図 9 には、各手法における計測エラー発生時の分類結果の平均 F 値を各データセットに対して算出したものを示している。表 2 には、各手法の計測エラー発生を分類した詳細な結果を示しており、それぞれの手法において、分類した 2 クラスとその平均の適合率、再現率、F 値を示している。

まず、過去の軌跡を用いる 3 つの位置推定器の結果に注目する。これらの中では、パーティクルフィルタの結果が最も良い。これは、参加者の移動速度や移動方向の頻繁な変化に対応し、計測エラーの発生を検出できたためである。しかし、通勤通学、地下街データセットにおいては、次のよ

表 2 各手法の計測エラーへの分類の詳細結果 (LWL)  
 Table 2 Classification performance for error detection (LWL).

	カルマンフィルタ			改良カルマンフィルタ			パーティクルフィルタ			他のプロバイダ		
	適合率	再現率	F 値	適合率	再現率	F 値	適合率	再現率	F 値	適合率	再現率	F 値
計測エラーなし	0.897	0.701	0.787	0.924	0.668	0.776	0.944	0.661	0.778	0.963	0.643	0.771
計測エラーあり	0.411	0.722	0.523	0.413	0.809	0.547	0.424	0.864	0.569	0.425	0.914	0.58
平均	0.654	0.711	0.655	0.668	0.739	0.661	0.684	0.763	0.673	0.694	0.779	0.676

	修正 Wi-Fi スキャン ( $g = 3$ )			修正 Wi-Fi スキャン ( $g = 5$ )			提案手法		
	適合率	再現率	F 値	適合率	再現率	F 値	適合率	再現率	F 値
計測エラーなし	0.973	0.57	0.719	0.973	0.57	0.719	0.97	0.923	0.946
計測エラーあり	0.388	0.944	0.55	0.388	0.944	0.55	0.772	0.901	0.832
平均	0.68	0.757	0.635	0.68	0.757	0.635	0.871	0.912	0.889

うな状況が頻繁に現れており、位置推定の精度を低下させていることが分かった。参加者がトンネルを通る場合や周囲に店の少ない地下通路を移動する場合は、参加者の周囲に AP が存在せず、長時間 Wi-Fi 位置計測を行えない。その場合は、Wi-Fi 位置計測が途切れる前の軌跡をもとに現在位置を推定するため、実際の位置と推定位置間の距離が数百 m 程度と大きくなってしまふ。その結果、計測エラーとして誤検出してしまったと考えられる。これは、他の 2 つの位置推定器についてもいえることである。対して、改良カルマンフィルタに注目すると、地下街データセットにおいてパーティクルフィルタの精度を上回っていることが分かる。これは、移動した地下通路は直線的に設計されることが多いため、移動データが直線的な移動を前提とする改良カルマンフィルタに適合したためだと考えられる。

次に、他のプロバイダへの問合せによる結果に注目する。図 8 から、この手法は他の位置推定器を用いた手法と比べて、良い結果が得られていることが分かる。しかし、この手法は参加者が大学や公園や駅といった広い面積のエリアを移動するとき、正確な位置推定を行うことができていなかった。

修正 Wi-Fi スキャンによる問合せによる結果に注目すると、通勤通学データセットにおける精度が悪いことが分かる。このとき  $g$  は、AP を分割するグループ数に対応する。通常、Wi-Fi 計測エラーは周囲の AP が少ない場合に発生しやすい。そのため、電車やバスへの乗車により、周囲の AP が少なくなる状況が発生する通勤通学データセットにおいては、この手法は AP のグルーピングが行えないために正確な位置推定が行えない。また、この手法は階層的クラスタリングを行い、AP のグループ数の過半数から構成されるクラスタを発見する。そして、そのクラスタに含まれる AP を用いて位置推定を行う。このとき通勤通学データセットのように、ユーザの周辺の異常 AP の数が多くなったとき、そのクラスタには多くの誤った推定位置が含まれてしまふ。すなわち推定位置の信頼性が低くなり、エラー検出に悪影響を及ぼす。一方で、正しい推定位置が大半を占めるクラスタの場合は、それらの位置の分散は小

さくなる特徴があると考えられる。この分散を分類器の特徴に含めることで、分類器がそのような特性を学習する可能性があると考えられる。

最後に提案手法に注目すると、他の手法と比べてきわめて良い精度で計測エラーを検出できていることが分かる。その平均 F 値は 0.889 であり、他手法と比べて約 0.2 も向上している。前述のように、各位置推定器は計測環境に応じて性能が変化するが、提案手法は計測環境の特徴量から、最適な位置推定器の意見を計測エラーの検出に反映させる。さらに、計測環境の特徴量を直接的に用いて計測エラーの検出に反映させることもできると考える。そのため、図 9 のように、3 種類のデータセットすべてにおいて、高い精度を得ることができていた。特に、通勤通学データセットに関して精度の向上が著しかった。この結果に関して、情報ゲインを用いたそれぞれの特徴量の分類タスクにおける貢献度の評価 [13] を行ったところ、移動速度の特徴が最も分類に貢献していた。すなわち、この結果では移動速度に関する計測環境の特徴が直接分類に貢献していた。また、ビル街のデータに関する精度が高くなっている。ビル街のデータには、ショッピングなどの大きな移動を含まない状況のデータも含まれる。本研究で用いる計測環境の特徴は、主にユーザの移動を前提としたものが多いため、このような状況におけるエラー検出に大きく寄与しない。しかし、そのような状況においては、カルマンフィルタやパーティクルフィルタによって高い精度での位置予測が可能のため、ビル街のデータの精度が高くなっていると考えられる。

また表 1 に示すとおり、それぞれのデータセットごとに用意した特徴ベクトルは約 200 程度である。また、それぞれのデータセットを用意するのに要した時間は 5 時間ほどである。今回の実験ではデータを取得していなかった移動媒体（たとえば自転車など）により得られたデータを認識する場合は、これだけの特徴ベクトルを学習用データとして用意すれば十分に高い精度を得ることができると考える。過去の履歴数による影響：ここまでの実験結果は、すべての過去の履歴が利用可能であるという前提のもとに得られたものである。しかし、実際には Wi-Fi 位置計測時に十分



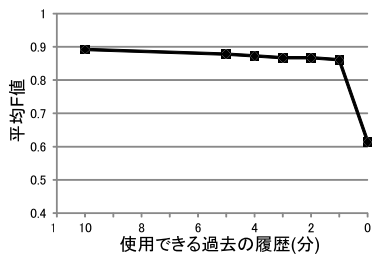


図 10 過去の履歴数による提案手法の分類精度の変化

Fig. 10 Relationship between average F-measure and quantity of history data to be used.

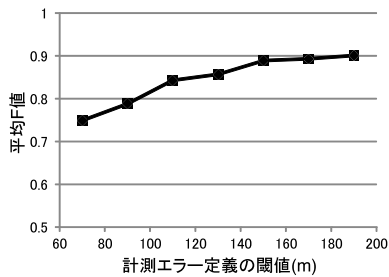


図 11 計測エラーの閾値と分類精度の変化

Fig. 11 Accuracies with changes of threshold of Wi-Fi positioning error.

過去の履歴が存在するとは限らない。たとえば、写真などへのメタデータとして位置情報を使用する場合や、ナビゲーションの開始直後では、過去の履歴は存在しないだろう。そのような場合の評価を行うため、使用する履歴を過去 0 分から 10 分まで変化させて、提案手法による計測エラーの検出を行った。その結果を図 10 に示す。本実験では 1 分ごとに Wi-Fi 位置計測を行っているため、履歴の 1 分が 1 組の計測データに対応する。図 10 より、過去の履歴をまったく使用しない場合、提案手法の検出精度が極端に低いことが分かる。これは、過去の履歴が存在しなければ使用できない位置推定器や計測環境の特徴があるためである。一方、1 分でも過去の履歴が存在すれば、提案手法は良い検出精度を得ており、これは、前述の過去の履歴が存在しないような場合でも、1 分間待つことで計測エラーの検出が可能であることを示している。

計測エラーの検出感度：ここでは、提案手法の計測エラーの検出感度について論じる。まず、提案手法がどれほど誤差の小さい計測エラーを検出できるかを検証する。これまでは計測エラーを定義する閾値を 150 m として計測エラーの検出を行ってきたが、図 11 に計測エラーを定義する閾値を変化させたときの、提案手法の平均 F 値の変化を示す。図 11 より、閾値を小さくすると検出精度が著しく低下することが分かる。対して、閾値を 150 m 以上としても検出精度に大きな変化は見られず、閾値を 150 m とすることで問題なく計測エラーの検出が可能であることを確認できる。

次に、提案手法の計測エラー検出における ROC 曲線を

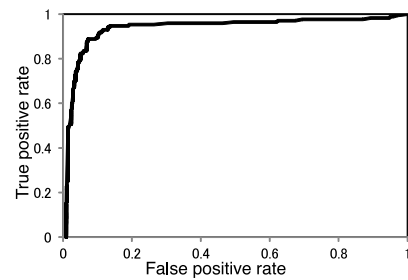


図 12 提案手法の ROC 曲線 (LWL)

Fig. 12 ROC curve for proposed method (LWL).

図 12 に示す。2 クラス分類器は、対象とするデータがどちらのクラスに分類されるかの確率を算出し、0.5 を超えている方のクラスへと分類を行っている。本実験においては、この分類の閾値を変化させることが、計測エラーの検出感度を変化させることに相当する。ROC 曲線は、この分類の閾値を変化させた際の結果をプロットしたものである。横軸の False positive rate は「計測エラーなし」のデータ中で「計測エラーあり」と誤分類された割合を、縦軸の True positive rate は「計測エラーあり」のデータ中で「計測エラーあり」と正確に分類された割合を示している。図 12 から、True positive rate を 0.95 とし、ほぼすべての計測エラーを検出できるように設定しても、False positive rate が約 0.2 と低い値を保っていることが分かる。

#### 4.2 Wi-Fi 位置計測における計測エラーの修正

評価手法：評価のために 4.1 節と同様に、leave-one-out 交差検定を行った。計測エラーの修正における誤差推定器が使用する回帰分析には、SMO 法 (Sequential Minimal Optimization) [14] を使用した。式 (1) の  $e_{nT}$  は線形回帰を用いて以下のように表される。

$$e_{nT}(f_T) = w \cdot \phi(f_T) + b$$

このとき、 $f_T$  は計測環境に関わる特徴を連結することで作成した特徴ベクトル、 $w$  は回帰係数ベクトル、 $b$  はバイアス項、 $\phi$  は特徴空間への写像であり実際にはカーネルとして多項式カーネルを用いている。これらのパラメータ推定手法の詳細に関しては、文献 [14] を参照されたい。また、入力とするデータは誤差が 150 m 以上の計測エラーが発生しているデータのみとした。さらに、提案手法の有効性を示すため、最も先進的なトラッキング手法の 1 つであるパーティクルフィルタを用いて位置計測 (修正) を行う手法の評価も行い、提案手法と比較する。パーティクルフィルタを用いた手法は、パーティクルフィルタにより予測された位置を修正位置として用いるものである。

評価結果・考察：図 13 は、修正前の Wi-Fi 計測位置と提案手法による修正位置のそれぞれの誤差の累積密度を示している。図 13 における、提案手法による修正位置の累積密度と修正前の Wi-Fi 計測位置の累積密度の違いから、提

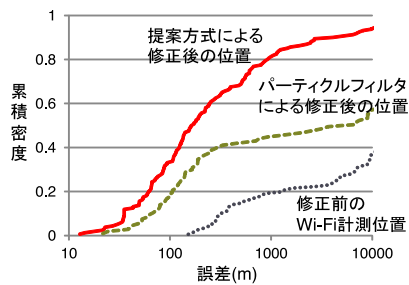


図 13 提案手法による修正前後の誤差の CDF

Fig. 13 CDFs of our method and original Wi-Fi positioning.

案手法は修正前の Wi-Fi 計測位置の誤差を多く減少させていることが確認できる。特に、提案手法による修正位置の誤差 150 m における累積密度は 0.475 となっている。これは、計測エラーが発生している Wi-Fi 計測位置のうち 47.5% を、計測エラーが発生していないと判断される誤差 150 m 未満へと修正できたことを意味する。通常の Wi-Fi 位置計測手法では表 1 に示すように、約 2 割の観測において 150 m 以上の誤差が確認されているため、誤差修正手法により 150 m 以上の誤差を 1 割にまで低減できることになる。また、パーティクルフィルタを用いた修正手法は 28.9% のエラーを 150 m 以内に修正できていたが、提案手法には及ばないことが分かる。上述のとおり、提案手法は様々な特性を持つ予測器を組み合わせることで用いることができるため、単一の予測器に比べて精度が高い。以上から、提案手法により Wi-Fi 位置計測に用いたデータのみを用いて、Wi-Fi 位置エラーの修正が行えたことを確かめ、提案手法の有効性を確認した。

### 5. 関連研究

Wi-Fi 位置推定を GPS と組み合わせることで省電力な位置計測を試みる研究は多くある。Lin ら [15] は特定のアプリケーションにおいて、ユーザが必要とする位置計測精度を見積もり、それに応じて適切な位置計測手法を選択することで電力消費を抑えている。一方で本研究は、Wi-Fi 位置計測の信頼性に応じて GPS と Wi-Fi の利用の切替えを可能にするものである。また、環境の状況を考慮して Wi-Fi 位置計測を行う研究もある。Chen ら [16] は、気温や湿度といった Wi-Fi 電波の伝搬に関係する指標を環境に設置したセンサネットワークにより取得し、適応的に位置推定を行っている。一方で本研究では、携帯端末から得られるコンテキスト情報（周囲の AP の数など）に加えて、位置予測器の結果も用いて Wi-Fi 位置計測の信頼性を推定している。さらに、Wi-Fi 位置計測における内部のデータベースの健全性を維持する研究も行われている。何ら [6] は、エンドユーザが収集した Wi-Fi スキャンデータを用いて破棄された AP や移設した AP を検知している。基本的には、新しく収集された Wi-Fi スキャンデータに、過去に観測された AP が含まれなかった場合、その AP が破棄・

移設されたと見なしている。藤木ら [5] は、さらに AP の電波の伝搬範囲を考慮した検知を行っている。一方で、本研究では Wi-Fi 位置計測の内部をブラックボックスとし、その出力にエラーが含まれているかどうかを識別する。

### 6. おわりに

本研究では、Wi-Fi 位置計測において異常 AP により発生する計測エラーを検出し、さらに計測エラー検出時の計測位置を修正する手法を提案した。Wi-Fi 計測エラーは様々な環境で起こりうるため、単一の位置推定器により正確に現在位置を推定することは困難であった。そこで、Wi-Fi 位置計測を行った周囲の環境を考慮して、様々な特徴を持つ位置推定器を複数組み合わせることで計測エラーの検出を行う手法を実現した。また、同様の位置推定器を用いることで、計測位置を修正する手法を実現した。評価実験では、実際に様々な状況を想定して収集した Wi-Fi 位置計測データを使用し、提案手法の有効性を検証した。

### 参考文献

- [1] Vaupel, T., Seitz, J., Kiefer, F., Haimerl, S. and Thielecke, J.: Wi-Fi Positioning: System Considerations and Device Calibration, *International Conference on Indoor Positioning and Indoor Navigation*, pp.1-7, IEEE (2010).
- [2] 暦本純一, 塩野崎敦, 末吉隆彦, 味八木崇: PlaceEngine: 実世界集合知に基づく WiFi 位置情報基盤, インターネットコンファレンス, Vol.2006, pp.95-104 (2006).
- [3] 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 (2007).
- [4] Jones, K. and Liu, L.: What where wi: An Analysis of Millions of Wi-Fi Access Points, *IEEE International Conference on Portable Information Devices, 2007, PORTABLE07*, pp.1-4, IEEE (2007).
- [5] 藤木慎太郎, 相田 仁: ユーザ参加型無線 LAN 位置測定システムにおけるアクセスポイント移設の検出法, 情報処理学会第 72 回全国大会講演論文集, Vol.3, pp.302-304 (2010).
- [6] 何 韜, 梶 克彦, 河口信夫: 位置推定のための無線 LAN 観測データベースの健全性維持手法, 情報処理学会第 73 回全国大会講演論文集, Vol.3, pp.281-282 (2011).
- [7] Deleted for blind review: Detecting and Correcting WiFi Positioning Errors, *UbiComp 2013* (2013).
- [8] Kang, J., Cohen, I. and Medioni, G.: Continuous Tracking within and Across Camera Streams, *CVPR 2003*, Vol.1, pp.267-272 (2003).
- [9] Doucet, A., De Freitas, N. and Gordon, N.: *Sequential Monte Carlo Methods in Practice*, Springer Verlag (2001).
- [10] Google Inc.: Android Developers Reference: Locationmanager, available from (<http://developer.android.com/>).
- [11] Zandbergen, P.: Accuracy of iPhone Locations: A Comparison of Assisted GPS, WiFi and Cellular Positioning, *Transactions in GIS*, Vol.13, No.s1, pp.5-25 (2009).
- [12] Skyhook Wireless, available from (<http://www.skyhookwireless.com/>).
- [13] Witten, I. and Frank, E.: *Data Mining: Practical Machine Learning Tools and Techniques*, Morgan

- Kaufmann (2004).
- [14] Shevade, S., Keerthi, S., Bhattacharyya, C. and Murthy, K.: Improvements to the SMO Algorithm for SVM Regression, *IEEE Trans. Neural Networks*, Vol.11, No.5, pp.1188-1193 (2002).
  - [15] Lin, K., Kansal, A., Lymberopoulos, D. and Zhao, F.: Energy-accuracy Trade-off for Continuous Mobile Device Location, *The 8th International Conference on Mobile Systems, Applications, and Services*, pp.285-298, ACM (2010).
  - [16] Chen, S., Chen, Y. and Trappe, W.: Exploiting Environmental Properties for Wireless Localization and Location aware Applications, *PerCom 2008*, pp.90-99 (2008).



津田 祐輝

平成 23 年大阪大学工学部電子情報工学科卒業。平成 25 年同大学院情報科学研究科博士前期課程修了。現在、株式会社 JSOL に勤務。モバイルコンピューティングに興味を持つ。



孔 全 (学生会員)

平成 23 年西安交通大学ソフトウェア学院ソフトウェアエンジニアリング専攻卒業。現在、同大学院情報科学研究科博士前期課程在学中。ユビキタスコンピューティングに興味を持つ。



前川 卓也 (正会員)

平成 15 年大阪大学工学部電子情報エネルギー工学科卒業。平成 18 年同大学院情報科学研究科博士後期課程修了。同年日本電信電話株式会社入社。平成 24 年 4 月より大阪大学大学院情報科学研究科准教授。博士 (情報科学)。本会平成 22 年度山下記念研究賞等受賞。ACM, 電気学会, 日本データベース学会各会員。