

Indoor Positioning Using Multi-Frequency RSS with Foot-Mounted INS

Martin Nilsson, Jouni Rantakokko, Martin A. Skoglund and Gustaf Hendeby

Linköping University Post Print



N.B.: When citing this work, cite the original article.

Original Publication:

Martin Nilsson, Jouni Rantakokko, Martin A. Skoglund and Gustaf Hendeby, Indoor Positioning Using Multi-Frequency RSS with Foot-Mounted INS, 2014, Fifth International Conference on Indoor Positioning and Indoor Navigation.

Copyright: ©2014 IEEE. Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works must be obtained from the IEEE.

Preprint available at: Linköping University Electronic Press

<http://urn.kb.se/resolve?urn=urn:nbn:se:liu:diva-111802>

Indoor Positioning Using Multi-Frequency RSS with Foot-Mounted INS

Martin Nilsson*, Jouni Rantakokko*[†], Martin A. Skoglund[‡], and Gustaf Hendeby^{‡§}

* Dept. of Robust Telecommunications, Swedish Defence Research Agency (FOI), Linköping, Sweden

[†] Dept. of Signal Processing, KTH Royal Institute of Technology, Stockholm, Sweden

[‡] Dept. of Electrical Engineering, Linköping University, Linköping, Sweden

[§] Dept. of Sensor & EW Systems, Swedish Defence Research Agency (FOI), Linköping, Sweden

Abstract—This paper presents a system which combines a zero-velocity-update-(ZUPT)-aided inertial navigation system (INS), using a foot-mounted inertial measurement unit (IMU), with opportunistic use of multi-frequency received signal strength (RSS) measurements. The system does not rely on maps or pre-collected data from surveys of the radio-frequency (RF) environment. Instead it builds its own database of collected RSS measurements during the course of the operation. New RSS measurements are continuously compared with the stored values in the database, and when the user returns to a previously visited area this can thus be detected. This enables loop-closures to be detected online and used for error drift correction. The system utilises a distributed particle simultaneous localization and mapping (DP-SLAM) algorithm which provides a flexible 2D navigation platform that can be extended with more sensors. The experimental results presented in this paper indicates that the developed RSS SLAM algorithm can, in many cases, significantly improve the positioning performance of a foot-mounted INS.

I. INTRODUCTION

A reliable and accurate positioning system is expected significantly improve the safety for first responders and to enhance their operational efficiency. To achieve this, a first responder positioning system must be able to provide at least room level accuracy during extended indoor operations, and in other *global navigation satellite system* (GNSS)-challenged environments. There are also other important requirements besides raw performance, such as weight and cost, to consider when designing a personal positioning system intended for safety-of-life applications. The system should be able to operate in unknown environments without relying on any pre-installed infrastructure [1]. An overview of the challenges associated with reliable indoor positioning for first responder applications, as well as sensor types that could be considered when implementing such systems, is given in [1].

Wearable mobile indoor positioning systems, intended for first responders and other pedestrian applications (such as location-based services, pervasive gaming, and navigation support), have received increased attention the last decade or so. A first approach to the problem is to use inertial sensors. With

the rapid progress of sensor miniaturization (*micro electro mechanical systems*, MEMS) technology, *inertial measurement units* (IMUs) have become both small and affordable enough to be used in pedestrian and professional applications. Examples of operational systems employ *pedestrian dead-reckoning* (PDR) type of algorithms are [2, 3]. In these the data from accelerometers, gyroscopes, magnetometers are used to detect when the person takes a step and to determine the orientation of the IMU. Motion classification and models can then be applied to decide upon the direction of the step.

Lately, many research groups have turned their focus towards foot-mounted IMUs, which has the potential to provide improved accuracy and robustness [1]. An early example using shoe-mounted inertial and magnetic sensors was presented in [4], and since then several similar systems have been developed [2, 5–8]. The key idea is to detect and utilize the stand-still phase of the foot during each gait cycle in a zero-velocity-update (ZUPT). This limits the drift caused by double integrating imperfect measurements, but cannot completely alleviate the effect which results in a degrade in the position estimate over time and/or travelled distance [8]. To obtain long time stable estimates, additional sensor information must be used.

One approach to improve the long term stability is to incorporate information about beforehand measured *received signal strength* (RSS) maps or floor plans to support the positioning. In [9] a method for acquiring RSS measurements to create a map is described and it is represented using *Gaussian processes* (GP). An early example of utilizing such map information is the RADAR system [11]. These methods have gained attention as they are suitable for implementation on smartphones, which have the required sensors. Relying on pre-collected map information works well where this is possible [12], *e.g.*, in shopping malls or office buildings [13]. However, first responder applications are more demanding as RSS maps cannot be obtained beforehand and, if they exist, are prone to change. In these situations *simultaneous localization and mapping* (SLAM) approaches can be employed instead.

There are several SLAM systems relying on RSS measurements, *e.g.*, WiFi GraphSLAM [14], an application of GraphSLAM [15] to use RSS measurements from WiFi; the GP based SLAM solution in [16]; and in [17] information from both previously known and unknown WiFi access points are

This work was funded by the Swedish Armed Forces through the Command and Control R&D program, reference AF9220209. The work has also been supported by the project Cooperative Localization (CoopLoc) funded by Swedish Foundation for Strategic Research (SSF), the Swedish strategic research center Security Link and the Vinnova Industry Excellence Center LINK-SIC

used for positioning. RSS measurements have also been used in conjunction with foot-mounted INS systems, in the WiSLAM method [18].

WiFi RSS measurements may fail in many first responder scenarios for several reasons, the electricity in the building may be lost or WiFi may not be installed at all. A less explored alternative is to employ multi-frequency RSS measurements of other opportunistic radio sources, *e.g.*, FM radio and TV stations. These are more or less available everywhere. The study presented in [19] indicates that a combination of FM and WiFi RSS measurements can be used to distinguish between different rooms with a high level of accuracy. This is collaborated by [20], which, however, also concludes that the RSS may vary significantly over time. A SLAM system utilizing opportunistic radio signals to navigate a small robot is described in [21], where the RSS map is modeled as a piece-wise linear function. A SLAM system, based on IMU, magnetometer, and FM RSS measurements and using a PDR, is described and evaluated in [22]. The system developed in this paper uses a similar approach, however, it integrates the multi-frequency RSS measurements with a foot-mounted INS which has the potential to provide higher accuracy and robustness in first responder applications. Furthermore, the presented RSS SLAM method differs from [22] as it incorporates a GP directly in the SLAM map to model the RSS field variation [23].

The system presented here adopts the occupancy grid based *distributed particle* SLAM (DP-SLAM) [24]. The occupancy grid allows for limiting the memory requirements of the algorithm, and the grid size provides a suitable positioning resolution. Using the occupancy grid also provides for an efficient way to determine if there are measurements available in a certain region of space.

This paper is organized the following way. After this introduction, the theory needed to perform RSS SLAM based on a ZUPT-aided foot mounted-system is provided in Sec. II, followed by a description of the experimental setup used to validate the approach in Sec. III. Results from the experiments are presented in Sec. IV, concluding remarks and a discussions of possible future work are provided in Sec. V.

II. THEORY

In this section the basic underlying concepts and theory which are used throughout the paper are presented.

A. SLAM

A general SLAM system can be described by the following equations

$$x_{k+1} = f(x_k, u_k, v_k), \quad (1a)$$

$$m_{k+1} = m_k, \quad (1b)$$

$$y_k = h(x_k, m_k) + e_k, \quad (1c)$$

where x_k is the state, m_k the map, u_k known or measured input, v_k process noise, y_k a measurement, and e_k measurement noise, all at time k . The model is divided into the motion model (1a), the map dynamics (1b), and the measurement model (1c).

A more in-depth explanation of the ZUPT-aided INS type motion model and the RSS measurement model is provided in this section. The map model follows the occupancy grid concept described below, where it is also described how new observations are added to the map at every iteration.

B. Map Representation

Distributed particle SLAM [24] is a particle filter based SLAM method which uses an ancestry tree together with an occupancy grid as map instead of landmarks. This way the ancestry of each particle and all its modifications to the grid can be tracked. When a particle is removed by resampling all its entries in the grid are also removed. When a particle is duplicated as a result of the resampling there is no need to copy the entire map over to the new particle. Instead, a new branch is created in the ancestry tree and the resampled particle and its siblings share all entries made by their ancestors in the occupancy grid made up to that point.

As the user moves in the environment and collects RSS measurements these are stored in an occupancy grid. Since the grid is combined with a particle filter this means that at every iteration the particle cloud is spread out across a number of cells. When a measurement update occurs, an RSS fingerprint from the user's true location will be recorded. Every particle will then enter the recorded fingerprint along with their identity, the current time and location into the grid cell which they currently occupy.

C. Motion Model

The pedestrian motion model is based on a ZUPT-aided INS [1] using a foot-mounted IMU. The state vector is given by $x_k = (p_x, p_y, \beta, \dot{\beta})^T$ and comprises: 2D position (p_x, p_y) , a heading bias β , and a heading bias rate $\dot{\beta}$. The motion model takes the input u_k , which is calculated using the current and the previous position estimates from the foot-mounted INS (using only position updates during consecutive zero-velocity instances). The input consist of the travelled distance and the heading angle

$$u_k = \begin{pmatrix} |p_{k+1}^{INS} - p_k^{INS}| \\ \angle(p_{k+1}^{INS}, p_k^{INS}) \end{pmatrix} = \begin{pmatrix} r_k \\ \phi_k \end{pmatrix}, \quad (2)$$

where $\angle(\cdot, \cdot)$ is the quadrant compensated arctangent function. The complete motion model is given by

$$x_{k+1} = x_k + \begin{pmatrix} r_k \cos(\phi_k - \beta) \\ r_k \sin(\phi_k - \beta) \\ T\dot{\beta} \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ \frac{T^2}{2} \\ T \end{pmatrix} v_k, \quad (3)$$

where $v_k \sim \mathcal{N}(0, Q)$ and T is the time difference between updates. The process noise compensates for errors in the INS estimate, its covariance, Q , should reflect typical INS error behavior and can as such be considered a tuning parameter.

The task of the particle filter is to constrain long term error drift, and it is necessary to include the position in the state vector since it is required for the grid mapping. The error accumulation model is based on the model for foot-mounted

IMUs suggested in [25]. Similar error accumulation behavior has been observed by others, see *e.g.*, [26]. The model used here is based on the following assumptions:

- All position errors are due to heading drift.
- The heading error can be modeled with constant growth rate.
- The heading error is a function of time, not distance.¹

D. Input From ZUPT-aided Foot-mounted INS

When walking or running the human foot is stationary for a short time when it is in contact with the ground. The idea behind zero-velocity updates is to detect this moment when the velocity of the foot is approximately zero and use this information to update the states in whatever positioning filter is being used. The moment the user's foot comes to rest (during the stance phase) and zero-velocity is detected, the 3D velocity vector, as well as the pitch and roll angles, becomes observable which can then be used to constrain the position error in the system. For a foot-mounted INS to work well there are some general guidelines:

- the foot should be on the ground often (stand still phase);
- a robust zero-velocity detection algorithm is needed; and
- performance depends on the user's movements, but to a lesser degree than for PDR-type systems.

In this work a foot-mounted IMU with three-axis accelerometers and gyroscopes is utilized. An *extended Kalman filter* (EKF) is used to estimate the user's position. A zero-velocity detection algorithm is then introduced which takes the gyroscope and accelerometer signals as inputs. There are several way of constructing zero-velocity detection algorithms but one which relies on both gyroscope and accelerometer data usually performs better than those relying on just one of these sensors [8]. Whenever zero-velocity is detected this is introduced as a pseudomeasurement in the EKF.

E. RSS Fingerprints

The RSS of electromagnetic signals in indoor environments are subject to rapid spatial variations due to multipath-induced fading. These spatial variations are caused by the electromagnetic signals interacting with the environment, which can be described in terms of well-known phenomena such as reflection, scattering, and diffraction [27]. This can however be exploited in positioning algorithms by using a technique called fingerprinting. An RSS fingerprint is a set of mean signal strength values collected at different frequencies. Generally a number of samples are recorded at each frequency and the mean values of the recorded sequences are stored in a vector. In short, if the mean RSS value at a specific frequency is Λ^i then the entire fingerprint vector is

$$\Lambda = (\Lambda^1 \quad \Lambda^2 \quad \dots \quad \Lambda^n)^T, \quad (4)$$

¹The error characteristics depends on the implementation. For instance, some implementations of foot-mounted INS do not allow the error to grow during a ZUPT. This assumption might thus be ill suited for such implementations, especially in scenarios where the user stands still for extended periods of time. A better model would then be to instead let the error grow with the travelled distance instead. Furthermore, the type of movement can significantly affect the error statistics.

where n is the total number of frequencies used in the fingerprint. A reason to use a set of n frequencies instead of just one is that RSS values from one frequency may be found at other locations. If a larger set of frequencies is used, then the chance of finding a unique fingerprint is increased.

Adding more frequencies to the set can improve the uniqueness of the fingerprint. However, blindly adding more frequencies is not guaranteed to improve the fingerprint and some care should be put into the selection process [20]. There are mainly two aspects to consider when choosing which frequencies to include in a fingerprint. The first being the frequency (and wavelength) of the radio signal as it directly affects how the signal interacts with objects in the environment. It is therefore a good idea to include signals from different frequency bands in the signal, as they will interact differently with the environment. The second important aspect to consider are the locations of the transmitters. If possible, signals should be included which are transmitted from different directions and distances relative to the operating area.

F. Gaussian Process Measurement Model

Each frequency in the RSS fingerprint vector is assumed to be a function of its position in 2D space which can be modelled with a GP. A GP is fully specified by its mean function $\mathcal{M}(x)$ and its covariance function $\mathcal{K}(x, x')$. The expression

$$\mathcal{F} \sim \mathcal{GP}(\mathcal{M}, \mathcal{K}), \quad (5)$$

should then be interpreted as the function \mathcal{F} is distributed as a GP with mean function \mathcal{M} and covariance function \mathcal{K} . Given a set of (possibly noisy) training samples from \mathcal{F} and a prior in the form of a mean function and covariance function, GPs can be used to infer information about unobserved function values. Let \mathcal{F} be the known training samples and let \mathcal{F}_* be the unobserved function values corresponding to the test inputs x_* . The joint distribution of \mathcal{F} and \mathcal{F}_* is

$$\begin{pmatrix} \mathcal{F} \\ \mathcal{F}_* \end{pmatrix} = \mathcal{N} \left(\begin{pmatrix} \mu \\ \mu_* \end{pmatrix}, \begin{pmatrix} \Sigma & \Sigma_* \\ \Sigma_*^T & \Sigma_{**} \end{pmatrix} \right), \quad (6)$$

where $\mu_i = \mathcal{M}(x_i)$, for $i = 1, \dots, n$, is the i th component of μ and analogously for μ_* . The Σ matrices is shorthand notation for the covariances between all combinations of test and training inputs. The predictive distribution of the unknown function values given the training samples becomes

$$\mathcal{F}_* | \mathcal{F} \sim \mathcal{N}(\mu_* + \Sigma_*^T \Sigma^{-1} (\mathcal{F} - \mu), \Sigma_{**} - \Sigma_*^T \Sigma^{-1} \Sigma_*). \quad (7)$$

The corresponding posterior process given the training data set \mathcal{D} with both inputs and observed function values is

$$\mathcal{F} | \mathcal{D} \sim \mathcal{GP}(\mathcal{M}_{\mathcal{D}}, \mathcal{K}_{\mathcal{D}}), \quad (8a)$$

$$\mathcal{M}_{\mathcal{D}}(x) = \mathcal{M}(x) + \mathcal{K}(x, x_*^T) \mathcal{K}(x, x')^{-1} (\mathcal{F} - \mu), \quad (8b)$$

$$\mathcal{K}_{\mathcal{D}}(x, x') = \mathcal{K}(x, x') - \mathcal{K}(x, x_*^T) \mathcal{K}(x, x')^{-1} \mathcal{K}(x, x_*), \quad (8c)$$

where the functional dependence on x and x' indicates that the posterior process can be computed for an arbitrary set of test cases. These expressions can also be found in *e.g.*, [10].

1) *Sparse Approximation Methods*: A well-known problem with Gaussian processes for on-line applications is the computational complexity. The main bottleneck is the inversion of the covariance matrix which has computational complexity $\mathcal{O}(n^3)$, where n is the number of data points used to construct the GP. One popular solution is to use a sparse approximation of the full GP and a number of available methods are compared and analyzed in [28]. This paper picks up on the ideas used in [29] where an assumption is made about the structure of the kernel function instead of the conditional distribution or the likelihood. Consider the radial basis kernel function (RBF)

$$\mathcal{K}(x, x') = \sigma_{\mathcal{F}}^2 \exp\left(-\frac{|x-x'|^2}{2l^2}\right). \quad (9)$$

This kernel function has local support which means that entries in the covariance matrix corresponding to measurement locations which are far away from each other will be close to zero. What is meant by far away is determined by the length scale parameter l in (9), which determines how quickly the correlation between points drop off.

If the correlation would be exactly zero, and not just close to zero after a certain distance, this would enable efficient implementations of GPs since only measurements from positions close to a predicted position would have to be included when constructing the GP. This can be achieved by using kernels with compact support. The RBF kernel in (9) can be compactified as

$$\mathcal{K}(x, x') = \sigma_{\mathcal{F}}^2 \exp\left(-\frac{|x-x'|^2}{2l^2}\right) \max\left(0, \left(1 - \left|\frac{x-x'}{d}\right|\right)^\nu\right), \quad (10)$$

where d defines the compact region over which the kernel has support and $\nu = 2\lfloor \frac{n}{2} \rfloor + 1$ where n is the dimension of the input to the GP. By compactifying the RBF kernel in this way it is guaranteed to remain positive definite [30]. By using (10) and only data which is inside the compact region the results will be almost as accurate as when also including data outside of the compact region.

2) *Hyper-parameter Selection*: Given a set of measurement values y , their corresponding positions x and a kernel function $\mathcal{K}(x, x')$, it is possible to make inference about the kernel's hyper-parameters. The marginal log-likelihood of the measurement values given their positions and the hyper-parameters is

$$L = \log p(y|x, \theta) = -\frac{1}{2} \log |K_y| - \frac{1}{2} y^T K_y^{-1} y - \frac{n}{2} \log(2\pi), \quad (11)$$

where K_y^{-1} is the covariance matrix produced using the kernel and the n data points and the mean function is assumed zero, $\mathcal{M}(x) \equiv 0$, for simplicity. It is worth to note that this expression balances model fit and complexity automatically. The first term is the complexity penalty term. The second term is the data-fit measure and it is the only term which depends on the measurement data y . The third term is a normalization constant which can be left out from the optimization as it does not depend on the data nor the hyper-parameters. It is now possible to find the hyper-parameter values that maximizes (11) using optimization routines.

3) *Measurement Model*: The measurement at time instant k is an RSS fingerprint vector, $y_k = (y_k^1 \dots y_k^n)^T$. Each component, $y_k^i = \Lambda_k^i + e_k^i$, contain an RSS value for a frequency as described in Sec. II-E and $e_k^i \sim \mathcal{N}(0, \sigma_e^2)$ is the measurement noise for component i . Using the current SLAM estimate, the predicted measurement is distributed according to

$$h(\hat{x}_{k|k-1}, \hat{m}_{k|k-1}^i) \sim \mathcal{N}(\hat{y}_k^i, \Sigma_k) \quad (12)$$

which is given by $\mathcal{GP}(\mathcal{M}_k^i, \mathcal{K}_k^i)$, where $\hat{x}_{k|k-1}$ and $\hat{m}_{k|k-1}^i$ are the predicted state and map for component i , respectively.

The compactified radial basis kernel function presented in Sec. II-F1 with added measurement noise is used to construct the GP covariance matrix

$$\mathcal{K}(x, x') = \sigma_{\mathcal{F}}^2 \exp\left(-\frac{|x-x'|^2}{2l^2}\right) \cdot \max\left(0, \left(1 - \left|\frac{x-x'}{d}\right|\right)^\nu\right) + \sigma_e^2 \delta_{xx'}, \quad (13)$$

where $\delta_{xx'}$ is the Dirac-delta function that is 1 if $x = x'$ and otherwise 0.

The GP is created from training data which here corresponds to previously recorded RSS values and their locations which are stored in the map database. The number of recorded values grows quickly during operation and it would be computationally intractable to use all of them in the GP. Instead the approximation in Sec. II-F1 is used and only the RSS fingerprints which are close to the RSS prediction are collected. The parameter d corresponds to the size of the kernel's support.

The mean function in the Gaussian process is chosen as the mean value of RSS values recorded in a region around the predicted location at each frequency. The hyper-parameters are pre-selected manually, to save computation time and to ensure feasible values. Given the GP it is straightforward to compute \hat{y}_k^i and Σ_k using (8b) and (8c), respectively. The likelihood of the recorded measurement y_k given the predicted state and map is

$$p(y_k | \hat{x}_{k|k-1}, \hat{m}_{k|k-1}) = \frac{\exp\left(-\frac{1}{2}(y_k - \hat{y}_k)(\Sigma_k + R_k)^{-1}(y_k - \hat{y}_k)^T\right)}{(2\pi)^{n/2} \sqrt{\det(\Sigma_k + R_k)}} \quad (14)$$

where n is the number of RSS measurements used in the fingerprint, and R_k is the covariance of the measurement noise.

III. EXPERIMENTAL SETUP

In this section, implementation details of the proposed RSS SLAM prototype are discussed and the experiments used to evaluate it are described.

A. Implementation Details

The proposed RSS SLAM algorithm has been implemented and used together with the hardware shown in Fig. 1. In the prototype implementation the ZUPT-aided foot-mounted INS in [7] is used, together with the FUNCube Dongle Pro+ radio



Fig. 1. Foot-mounted INS and antenna mounted on right leg of a user about to perform one of the experiments.

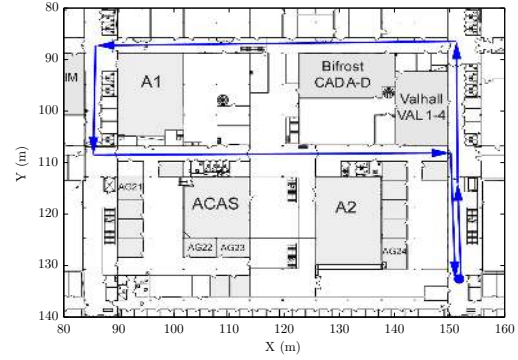


Fig. 2. Outline of the short loop-closure track.

receiver [31] and an omnidirectional antenna. A position estimate is extracted every time a ZUPT is registered in the foot-mounted INS, however only those in time with incoming RSS fingerprints are used to perform the DP-SLAM time updates. The filter update time delay is thus variable and bounded by the RSS fingerprint sample rate, which is set to 5 Hz, but also by whether or not ZUPTs are detected. To compensate for INS errors, the process noise was set to $Q = 0.0005 \text{ rad}^2/\text{s}^4$.

RSS measurements were derived by sampling the signal for seven different frequencies (94.4 MHz, 95.5 MHz, 97.3 MHz, 99.8 MHz, 103.2 MHz, 106.9 MHz, and 474.0 MHz) one by one at 192 kHz and then averaging the result over 4 ms. The covariance of the measured RSS was set to $R = \sigma_e^2 I$. To model the RSS variations (the map), the GP was tuned with the hyper-parameters $\sigma_{\mathcal{F}} = 8 \text{ dBm}$, $l = 4 \text{ m}$, $\sigma_e = 1.5 \text{ dBm}$, and $d = 8 \text{ m}$.

The DP-SLAM algorithm used to solve the SLAM problem was implemented as described in Sec. II using the parameters given above. The proposal distribution in the particle filter was chosen as $q(x_{k+1}|x_k, y_{k+1}) = p(x_{k+1}|x_t, u_t)$. To avoid applying the expensive resampling step at each iteration of the particle filter, the DP-SLAM implementation only performs resampling when necessary using sampling importance sampling (SIS). When to resample is determined by the estimated effective sample size, \hat{N}_{eff} , and resampling is only performed when $\hat{N}_{\text{eff}} < \frac{2}{3}N$, where N is the number of particles used in the filter, as suggested in [32]. Furthermore, $N = 1000$ is used as a compromise between computational complexity and accuracy.

B. Experiments

The A-house at Campus Valla, Linköping University, was chosen as place to conduct the experiments. The A-house was chosen as a suitable test facility given that it is fairly large and allows for many different ways to revisit the same place while providing strict pathways to stick to.

The algorithm is evaluated using three different tracks. All three tracks started and ended in the same point. A 5 m straight line at the start/end was provided for the user to walk on. The line allows for evaluating both deviations in position and

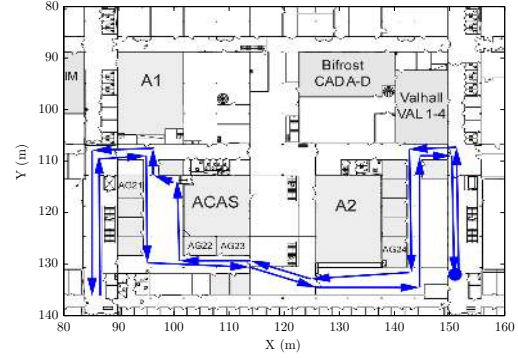


Fig. 3. Outline of the walked trajectory with backtracking.

heading at the end of the tracks. Note, however, that no ground truth exists for any other parts of the track, so it is necessary to resort to visual inspection comparing the solutions with and without the SLAM algorithm and with the approximate route on the map. Three different tracks were chosen; a short track with loop-closure; a track where the user walks back in his own foot-steps; and a longer track. All experiments were carried out with the sensor package (IMU and antenna) mounted on the right foot/leg as depicted in Fig. 1.

The first track, the short track with a loop-closure is depicted in Fig. 2, and it is about 220 m long and it takes the test subject approximately 3 min to walk. The track was designed to test the algorithms ability to perform loop-closure. Data was collected from walking the track ten times.

The second track, the track with backtracking approximately half way through, is depicted in Fig. 3. This track is a bit longer than the first one, approximately 340 m, resulting in a 4 min walk and it was repeated four times. This trajectory was chosen to observe how the algorithm deals with immediately revisiting a mapped region as a result of going back in the same footsteps after the full turn half way through the track.

The third track is almost three times as long and intended to evaluate drift over time. The track is depicted in Fig. 4 and it is 1100 m long and it takes 13 min to walk through it. This track was only walked once.

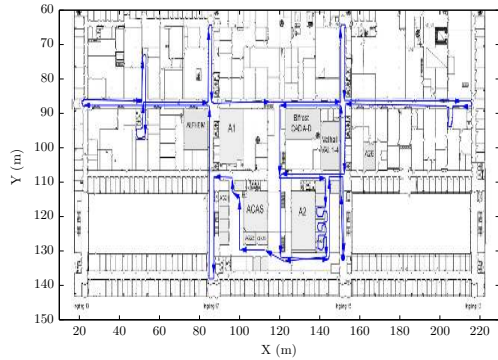


Fig. 4. Outline of the walk backtrack.

IV. RESULTS

In this section the results of the experiments described in Sec. III-B are presented and evaluated. Furthermore, quantitative results are presented to illustrate the effect of changing the GP hyper-parameters, as well as what happens if the number of particles are changed.

Overall, the behavior of the underlying foot-mounted INS can in these experiments be categorized as:

- *very slowly drifting*, low position and heading errors at the end of the track and a plausible trajectory throughout the experiments;
- *slowly drifting*, moderate position and heading errors at the end of the track and a roughly correct trajectory; and
- *heavily drifting*, considerable position and heading errors at the end of the track and an obviously skewed trajectory².

A. Short track with loop-closure

The results from the first trajectory (a short track with loop closure) are used to illustrate the different INS behaviors (see Fig. 5–6).

First consider the case when the foot-mounted INS system performs well, as illustrated in Fig. 5. In this case the RSS SLAM solution follows that of the foot-mounted INS closely, as expected, and only small adjustments are made to improve the estimates on loop-closure.

The typical foot-mounted INS system behavior, a slow drift, is illustrated in Fig. 6. In this case the INS estimate and the RSS SLAM estimate slowly drift apart, and the particle cloud grows with time. Once loop-closure is detected, the SLAM estimate is rectified to reflect the fact that a position is revisited. At the same time the particle cloud shrinks to reflect the fact that the position is better known after loop-closure.

Finally, Fig. 7 illustrates the behavior of the system in the case of heavy drift in the foot-mounted INS. The RSS SLAM estimate initially follows the INS estimate, and as it drifts off

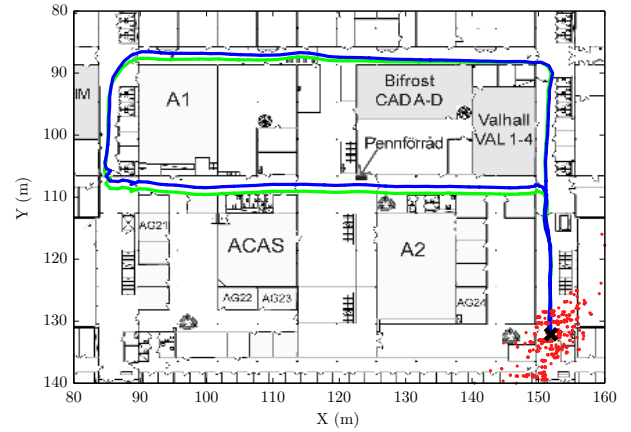


Fig. 5. Estimated trajectories using only foot-mounted INS (green) and with RSS SLAM (blue). In this case the INS system is well behaved.

TABLE I
SHORT LOOP-CLOSURE TRACK PERFORMANCE RESULTS AT THE END OF THE TRAJECTORY. EACH ROW IS DERIVED FROM 10 EXPERIMENTS.

(a) Position errors given in meters.

| RSS SLAM | Mean | Median | Std. dev | Max | Min |
|----------|------|--------|----------|------|-----|
| No | 8.9 | 7.3 | 6.4 | 22.1 | 1.9 |
| Yes | 2.6 | 1.9 | 2.7 | 9.2 | 0.5 |

(b) Heading errors given in degrees.

| RSS SLAM | Mean | Median | Std. dev | Max | Min |
|----------|------|--------|----------|------|-----|
| No | 14.2 | 9.4 | 9.9 | 31.9 | 4.8 |
| Yes | 2.0 | 1.4 | 2.1 | 7.2 | 0.1 |

the RSS SLAM particle cloud struggles to support the true position. This is visible already in the top figure in Fig. 7. This indicates an underestimated process noise and the case in the illustration is an outlier in term of INS behavior. The situation grows worse as time progress; however, when the loop-closure occurs a few particles are available to support it. Consequently, the RSS SLAM trajectory is improved compared to the foot-mounted INS estimate.

All the experiments performed with the short track with loop-closure are summarized in Table I. Both the median and mean errors indicate significant improvements when using DP-SLAM. The standard deviation and maximum of the errors are also significantly lowered for both heading and final position estimates, which indicate an increase in robustness.

B. Track with Back-Tracking

A typical result from the track with backtracking is presented in Fig. 8. Initially the foot-mounted INS and the SLAM algorithm yield similar results, but as the user turns and start to make his way back the SLAM algorithm starts to recognize the previously visited radio environment. This results in loop-closure, which reduces the uncertainty (the spread of the particle cloud). The resulting SLAM estimate seems to converge to a trajectory estimate which visually seems accurate when compared to the trajectory outline in Fig. 3.

²Analysis of the raw INS data from these cases indicate that the large errors are due to the gyro bias drifting during the walk, which is in turn likely due to warm-up effects in the IMU [23]. The INS implementation used here only estimates gyroscope bias at start-up.

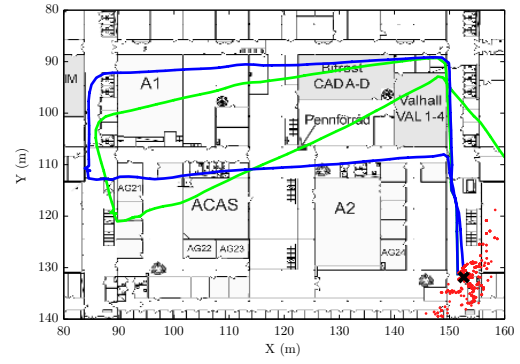
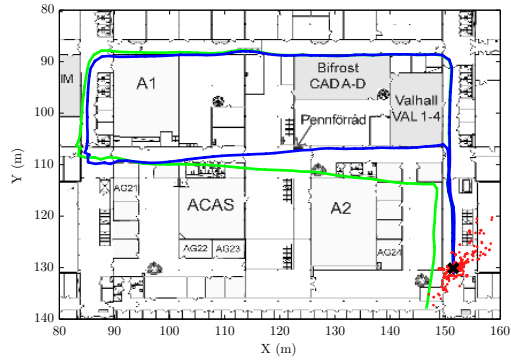
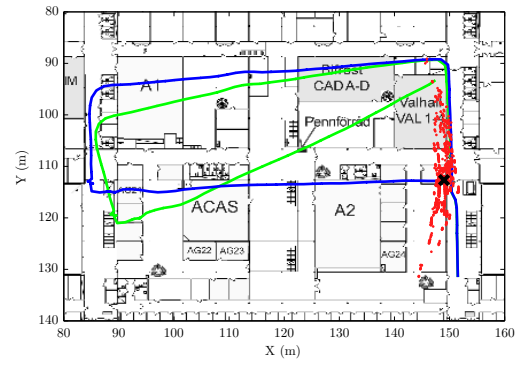
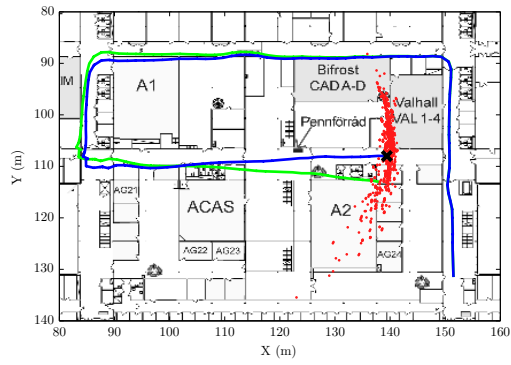
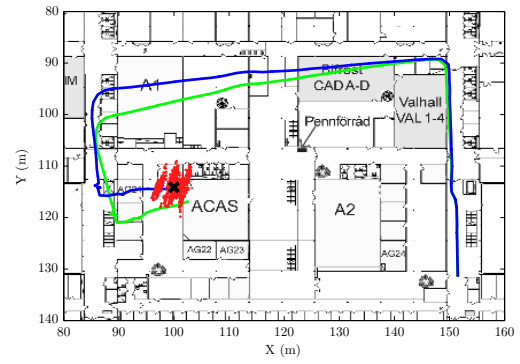
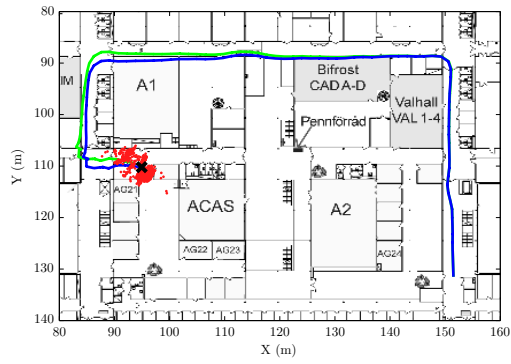
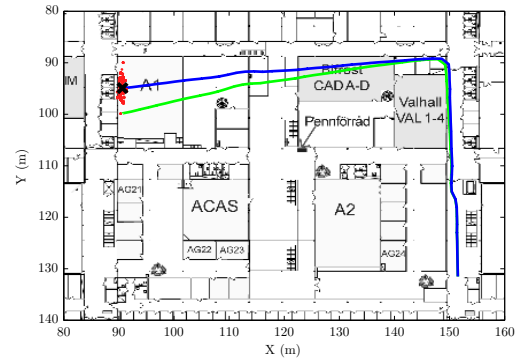
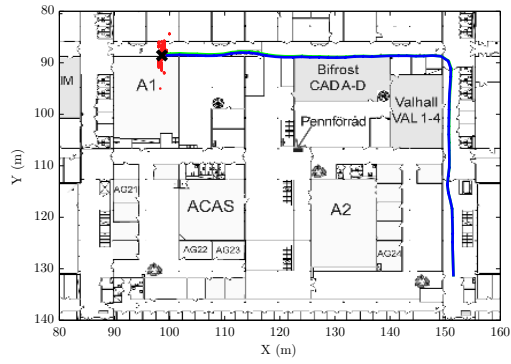


Fig. 6. Trajectories estimates using only foot-mounted INS (green) and with RSS SLAM (blue). Typical behavior, the INS drifts slowly.

Fig. 7. Trajectories estimates using only foot-mounted INS (green) and with RSS SLAM (blue), when the INS drifts heavily.

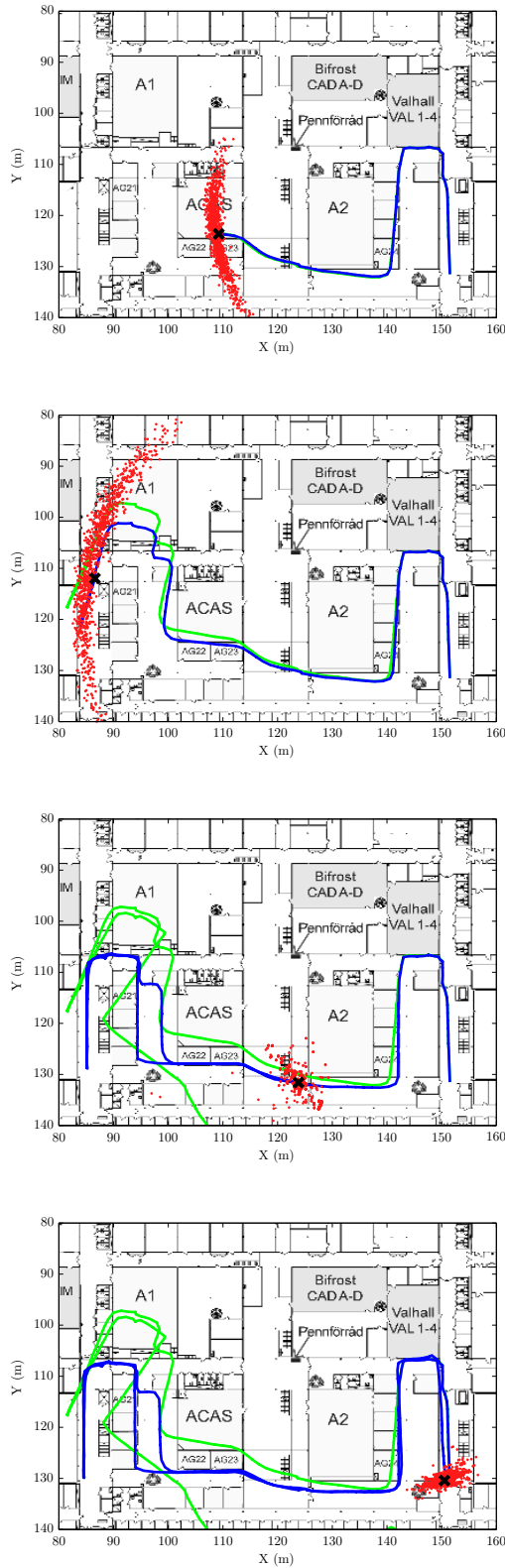


Fig. 8. Representative estimated trajectories using only foot-mounted INS (green) and with RSS SLAM (blue) performing the track with backtracking.

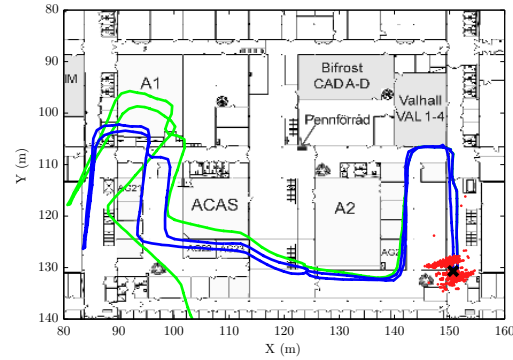


Fig. 9. Estimated trajectories using only foot-mounted INS (green) and with RSS SLAM (blue) performing the track with backtracking, for a case with worse than typical performance.

TABLE II
WALK BACK TRACK PERFORMANCE RESULTS. EACH ROW BASED ON 4 EXPERIMENTS.

(a) Position errors given in meters.

| RSS SLAM | Mean | Median | Std. dev | Max | Min |
|----------|------|--------|----------|------|------|
| No | 32.2 | 32.8 | 10.4 | 43.5 | 19.8 |
| Yes | 2.0 | 2.0 | 0.2 | 2.2 | 1.9 |

(b) Heading errors given in degrees.

| RSS SLAM | Mean | Median | Std. dev | Max | Min |
|----------|------|--------|----------|------|------|
| No | 54.7 | 53.5 | 19.0 | 76.5 | 35.4 |
| Yes | 2.9 | 3.1 | 1.9 | 5.0 | 0.5 |

Fig. 9 represents a less successful experiment, but the final DP-SLAM estimate is still a significant improvement compared to the foot-mounted INS. In this case the INS result drifts a little more when compared to the case in Fig. 8, this minor increase in INS drift probably causes the slightly worse trajectory estimate.

Overall, using the suggested SLAM significantly improves the trajectories obtained from the pure foot-mounted INS. However, it seems that this type of scenario is a more difficult challenge than the short loop-closure scenario when it comes to estimating the entire trajectory correctly. The results from the experiments are summarized in Table II. The improvement in heading and position errors are excellent, which is expected in this scenario since the person moves a long time in previously mapped areas.

C. Long Track

The trajectories estimated based on the long trajectory data is depicted in Fig. 10. The results are not as good as for the two shorter scenarios; however, the results are still promising. The foot-mounted INS drifts considerably over time, and the SLAM algorithm fails to completely compensate for this, but the improvement is still significant for parts of the test. The trajectory obtained from the SLAM system can still be identified as the true trajectory even if is not perfectly aligned with the floor plan.

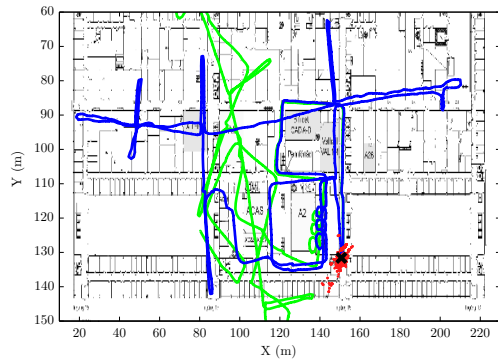


Fig. 10. Estimated trajectories using only foot-mounted INS (green) and with RSS SLAM (blue) for the long track and $l = 4$.

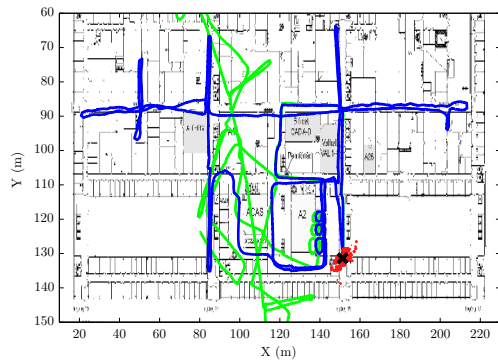


Fig. 11. Estimated trajectories using only foot-mounted INS (green) and with RSS SLAM (blue) for the long track and $l = 3$.

D. Gaussian Process Hyper-Parameters

The GP hyper-parameters used when modelling the map in the RSS SLAM algorithm is one source of uncertainty. Therefore, the experimental data has been evaluated using different parameter sets to quantify the difference. Tweaking the hyper-parameters results in no significant difference in the results obtained with the two shorter scenarios. For the last, longer, scenario a slight performance improvement can be observed by reducing the length-scale parameter from $l = 4$ m to $l = 3$ m. This improvement is illustrated in Fig. 11.

E. Number of Particles

The number of particles used in a particle filter is an important factor for its performance. Therefore, two data sets from the short track with loop-closure (one with low INS drift and one with large drift), and one data set from the track with backtracking, were selected to evaluate the importance of the number of particles used in the proposed RSS SLAM algorithm. The algorithm was applied 100 times for each number of particles to avoid random effects in the particle filter. The result of the investigation is shown in Fig. 12. The results indicate that in all of the tested scenarios the particle count could be lowered without taking a big toll on heading estimation performance. These results also indicate

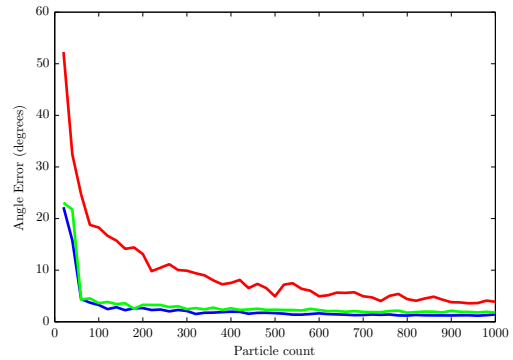


Fig. 12. Particle count effect on heading error estimation for typical short-loop closure scenario (blue), quickly diverging short-loop closure scenario (red) and walk-back scenario (green).

that performance in some scenarios are more sensitive to the particle count. Judging from these results 200 particles seems to be sufficient in favorable scenarios, while up to 600 particles may be required in other scenarios.

V. CONCLUSIONS AND FUTURE WORK

A new online capable algorithm for performing opportunistic radio received signal strength simultaneous localization and mapping (RSS SLAM) in indoor environments have been developed. It incorporates Gaussian processes to iteratively model the surrounding radio environment as it is being explored. The algorithm has been evaluated together with a foot-mounted INS and the experimental results indicate that it can successfully constrain the error drift of such systems where the error usually grows with time, as long as previously visited environments are re-visited during the operation. The algorithm does not rely on any pre-installed infrastructure or survey data and the radio transmitter locations does not need to be known.

There are aspects that need to be investigated further such as the GP hyper-parameter values. So far these have been selected manually, but ideally they should be learned during operation. A possible solution could be to, at regular intervals, pick the most likely particle and solve the optimisation problem with its corresponding data set and then apply the same hyper-parameters to all particles.

The results indicate that it is possible to reduce the heading error from a foot-mounted INS by up to 80%, and the final position error by up to 70%, in simple scenarios using this approach. In favourable scenarios, with a lot of back-tracking, the heading and final position errors can be reduced by as much as 93%–95% when compared to those of a lone foot-mounted INS.

Another aspect that needs to be investigated further is the particle count. The preliminary results indicate that the filter should be able to perform reasonably well with a much smaller particle count.

In addition, there are more areas which should be explored before this algorithm can be a permanent part of a robust first responder indoor positioning system such as:

- Add compatibility with other movement types than walking, such as running and crawling, and investigate in what ways the error model needs to be changed [23].
- Enable 3D tracking. This would enable the filter to make corrections in height and it might be a good idea to also add pitch error as a state in the filter since the height estimate from the foot-mounted INS drifts slowly.
- Implement with a dual foot-mounted INS. This could require a new error model as the heading error might no longer be as dominant as it is in the walking case.
- Add a magnetometer to the system. The magnetic field could be included in the fingerprint vector similar to the way another radio frequency can be added.

ACKNOWLEDGMENT

The authors would like to thank Joakim Rydell, Erika Bilock and Patrik Eliardsson at the Swedish Defence Research Agency for fruitful discussions and input during the course of this work.

REFERENCES

- [1] J. Rantakokko, J. Rydell, P. Stromback, P. Handel, J. Callmer, D. Tornqvist, F. Gustafsson, M. Jobs, and M. Gruden, "Accurate and reliable soldier and first responder indoor positioning: multisensor systems and cooperative localization," *IEEE Transactions on Wireless Communications*, vol. 18, no. 2, pp. 10–18, Apr. 2011.
- [2] P. D. Groves, G. W. Pulford, C. A. Littlefield, D. L. J. Nash, and C. J. Mather, "Inertial navigation versus pedestrian dead reckoning: Optimizing the integration," in *20th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS)*, 2007, pp. 2034–2055.
- [3] A. Jimenez, F. Seco, C. Prieto, and J. Guevara, "A comparison of pedestrian dead-reckoning algorithms using a low-cost MEMS IMU," in *IEEE International Symposium on Intelligent Signal Processing (WISP)*, 2009, pp. 37–42.
- [4] E. Foxlin, "Pedestrian tracking with shoe-mounted inertial sensors," *IEEE Computer Graphics and Applications*, vol. 25, no. 6, pp. 38–46, 2005.
- [5] J.-O. Nilsson, I. Skog, P. Handel, and K. V. S. Hari, "Foot-mounted INS for everybody—an open-source embedded implementation," in *IEEE/ION Position Location and Navigation Symposium (PLANS)*, 2012, pp. 140–145.
- [6] S. Godha, G. Lachapelle, and M. E. Cannon, "Integrated GPS/INS system for pedestrian navigation in a signal degraded environment," in *Proceedings of the 19th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS)*, Sep. 2006, pp. 2151–2164.
- [7] J. Rantakokko, E. Emilsson, P. Stromback, and J. Rydell, "Scenario-based evaluations of high-accuracy personal positioning systems," in *IEEE/ION Position Location and Navigation Symposium (PLANS)*, 2012, pp. 106–112.
- [8] I. Skog, P. Handel, J.-O. Nilsson, and J. Rantakokko, "Zero-velocity detection—an algorithm evaluation," *IEEE Transactions on Biomedical Engineering*, vol. 57, no. 11, pp. 2657–2666, 2010.
- [9] J. Fink and V. Kumar, "Online methods for radio signal mapping with mobile robots," in *2011 IEEE International Conference on Robotics and Automation (ICRA)*, Shanghai, May 2011.
- [10] C. E. Rasmussen and C. K. I. Williams, *Gaussian processes for Machine Learning*. MIT Press, 2006.
- [11] P. Bahl and V. N. Padmanabhan, "RADAR: An in-building RF-based user location and tracking system," in *Proceedings of 19th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)*, Tel Aviv, Israel, Mar. 2000.
- [12] T.-N. Lin and P.-C. Lin, "Performance comparison of indoor positioning techniques based on location fingerprinting in wireless networks," in *International Conference on Wireless Networks, Communications and Mobile Computing*, vol. 2, Jun. 2005, pp. 1569–1574 vol.2.
- [13] "Senionlab," <http://www.senionlab.com>, accessed: Jul., 24, 2014.
- [14] J. Huang, D. Millman, M. Quigley, D. Stavens, S. Thrun, and A. Agrawal, "Efficient, generalized indoor WiFi GraphSLAM," in *2011 IEEE International Conference on Robotics and Automation (ICRA)*, Shanghai, May 2011.
- [15] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents series)*. MIT Press, 2005.
- [16] B. Ferris, D. Fox, and N. Lawrence, "WiFi-SLAM using Gaussian process latent variable models," in *Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI)*, Hyderabad, India, Jan. 2007.
- [17] P. Addesso, L. Bruno, and R. Restaino, "Integrated rss from unknown access points in wlan positioning," in *2011 7th International Wireless Communications and Mobile Computing Conference (IWCMC)*, Istanbul, Turkey, Jul. 2011, pp. 607–612.
- [18] L. Bruno and P. Robertson, "WiSLAM: improving FootSLAM with WiFi," in *2011 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, Guimaraes, Portugal, 2011.
- [19] Y. Chen, D. Lymberopoulos, J. Liu, and B. Priyantha, "FM-based indoor localization," in *Proceedings of the 10th international conference on Mobile systems, applications, and services*. ACM, 2012, pp. 169–182.
- [20] P. Eliardsson, U. Uppman, and J. Rantakokko, "Received signal strength measurements utilizing signals of opportunity — a feasibility study on the use of RSS for loop-closure in multisensor indoor positioning systems," FOI, Tech. Rep. FOI-R–3841–SE, Feb. 2014.
- [21] J.-S. Gutmann, E. Eade, P. Fong, and M. E. Munich, "Vector field SLAM—localization by learning the spatial variations of continuous signals," *IEEE Transactions on Reliability*, Jun. 2012.
- [22] R. Faragher, C. Sarno, and M. Newman, "Opportunistic radio SLAM for indoor navigation using smartphone sensors," in *2012 IEEE/ION Position Location and Navigation Symposium (PLANS)*, Apr. 2012, pp. 120–128.
- [23] M. Nilsson, "Indoor positioning using opportunistic multi-frequency RSS with foot-mounted INS," Master's thesis, Linköping University, 2014.
- [24] A. Eliazar and R. Parr, "DP-SLAM: Fast, robust simultaneous localization and mapping without predetermined landmarks," in *International Joint Conference on Artificial Intelligence*, vol. 3, 2003, pp. 1135–1142.
- [25] S. Wan and E. Foxlin, "Improved pedestrian navigation based on drift-reduced MEMS IMU chip," in *Proceedings of the 2010 International Technical Meeting of The Institute of Navigation*, 2010, pp. 220–229.
- [26] J. Rantakokko, P. Strömbäck, E. Emilsson, and J. Rydell, "Soldier positioning in GNSS-denied operations," in *Proc. of Sensors and Electronics Technology Panel Symposium (SET-168) on Navigation Sensors and Systems in GNSS Denied Environments*, 2012, pp. 1–12.
- [27] L. Ahlin and J. Zander, *Principles of Wireless Communications*, 2nd ed. Studentlitteratur AB, 1998.
- [28] J. Quinero-Candela, C. E. Rasmussen, and C. K. Williams, "Approximation methods for Gaussian process regression," *Large-scale kernel machines*, pp. 203–223, 2007.
- [29] A. Ranganathan, M.-H. Yang, and J. Ho, "Online sparse gaussian process regression and its applications," *IEEE Transactions on Image Processing*, vol. 20, no. 2, pp. 391–404, 2011.
- [30] B. Hamers, J. A. Suykens, and B. De Moor, "Compactly supported RBF kernels for sparsifying the gram matrix in LS-SVM regression models," in *Artificial Neural Networks—ICANN 2002*. Springer, 2002, pp. 720–726.
- [31] "Funcube dongle," <http://www.funcubedongle.com>, accessed: Jul., 31, 2014.
- [32] F. Gustafsson, *Statistical Sensor Fusion*, 2nd ed. Studentlitteratur AB, May 2013.