

SLAM Dance

Inertial-Based Joint Mapping and Positioning for Pedestrian Navigation

PATRICK ROBERTSON, MICHAEL ANGERMANN

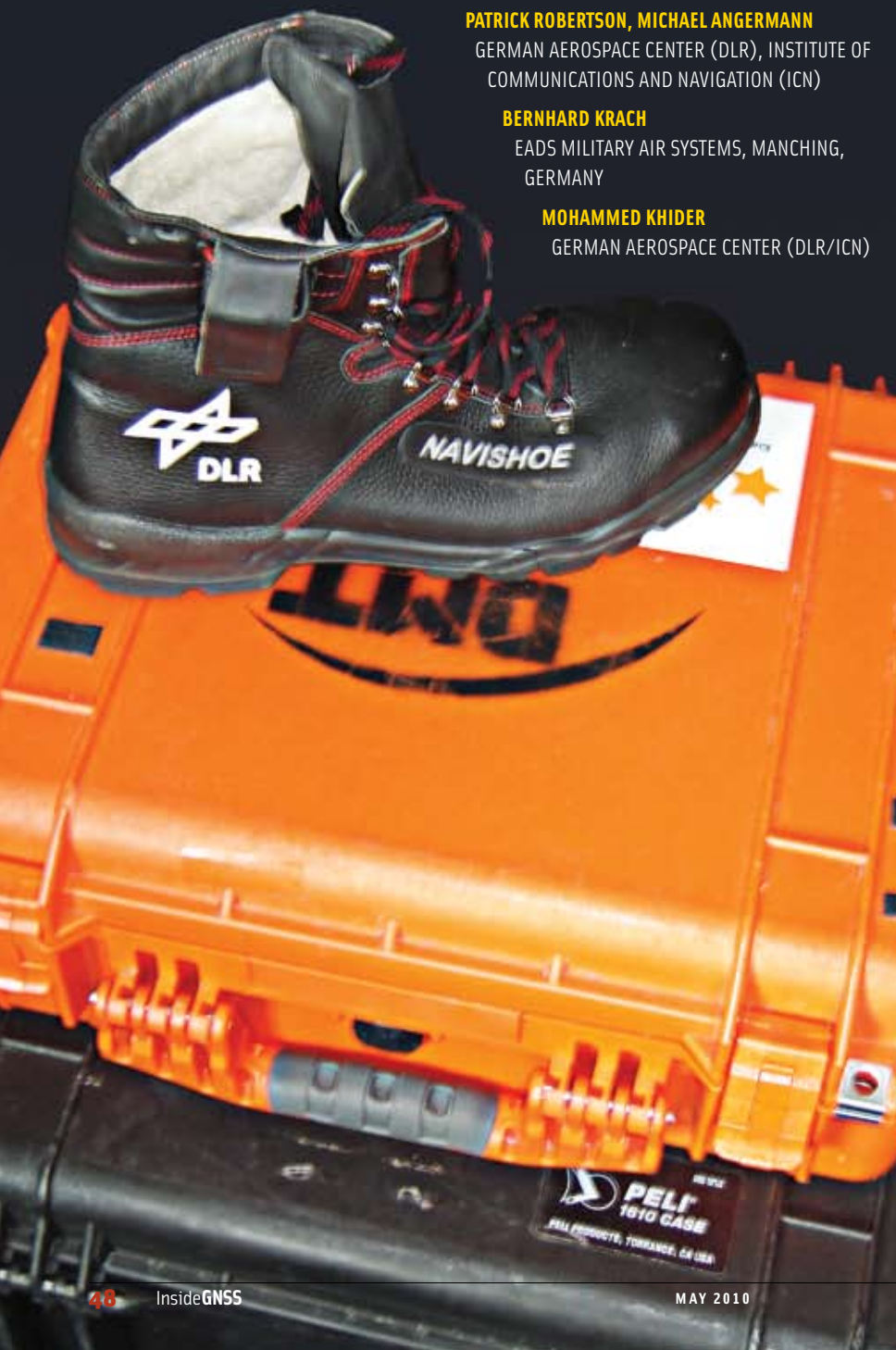
GERMAN AEROSPACE CENTER (DLR), INSTITUTE OF COMMUNICATIONS AND NAVIGATION (ICN)

BERNHARD KRACH

EADS MILITARY AIR SYSTEMS, MANCHING, GERMANY

MOHAMMED KHIDER

GERMAN AEROSPACE CENTER (DLR/ICN)



Imagine walking into a strange place – a shopping mall or an office building – wandering around for 20 minutes or so, and then coming out with a map of the facility that others could use to navigate through it – say, a fire rescue crew or simply someone looking for an office suite. A team of researchers at the German Aerospace Center are working on a “walk-about” solution that uses GPS to initialize the beginning of the traverse and tie the position data to an absolute coordinate frame. Then, foot-mounted inertial sensors increase the map accuracy when a person revisits previous points in the building.

Digital cartography and automated mapping techniques based on GNSS positioning have transformed our relationship to the physical world. The convergence of these complementary technologies are supporting the growth of commercial and consumer location-based applications that benefit from the coupling of real-time information with maps that are more current than ever – at least in environments that have access to radio signals from orbiting GNSS satellites.

Buildings, roads, mobile assets, points of interest, and people can be located outdoors based directly on GNSS-derived geographic coordinates or conventional addresses tied to these coordinates. Such advances, however,

are largely denied to us in underground or indoor venues where satellites signals do not reach.

Many prospective location-based service (LBS) applications — including safety-critical needs for emergency and security services — would become feasible if the associated mapping and real-time positioning requirements could be met. Finding alternative technologies that can meet these challenges have drawn the attention of many researchers and system developers.

Recent work has shown remarkable advances in the area of pedestrian indoor positioning aided by low-cost microelectro-mechanical system (MEMS) inertial sensors. At present, however, fully autonomous inertial navigation is still far from the realm of possibilities, due to sensor error-induced drift that causes position errors to grow unbounded within a few seconds.

This article introduces a new pedestrian localization technique that builds on the principle of simultaneous localization and mapping (SLAM). Our approach is called FootSLAM because it depends largely on the use of shoe-mounted inertial sensors that measure a pedestrian's steps while walking.

In contrast to SLAM used in robotics, our approach does not require specific feature-detection sensors, such as cameras or laser scanners. The work extends prior work in pedestrian navigation that uses known building plan layouts to constrain a location-estimation algorithm driven by a stride-estimation process. In our approach, building plans (i.e., maps) can be learnt automatically while people walk about in a building, either directly to localize this specific person or in an offline fashion in order to provide maps for other people.

We have combined our system with GPS and have undertaken experiments in which a person enters a building from outside and walks around within this building. The GPS position at the entry to the building provides a point of beginning for subsequent positioning/mapping without GPS.

Our experiments were undertaken by recording the raw sensor data and ground truth reference information. Offline processing and comparison with the ground-truth reference information allows us to quantitatively evaluate the achieved localization accuracy.

Building on the Past

The work of E. Foxlin cited in the Additional Resources section at the end of this article describes how we can use foot-mounted inertial measurement units (IMUs) to provide zero velocity updates — ZUPTs — during the rest phase of a pedestrian's stride with which to solve the problem of non-linear error growth over time with the aid of a Kalman filter. This is because an inertial navigation system (INS) can use the ZUPTs to accurately compute the displacement of the foot during a single step before errors would start to grow.

The zero update tells us when the step has been completed

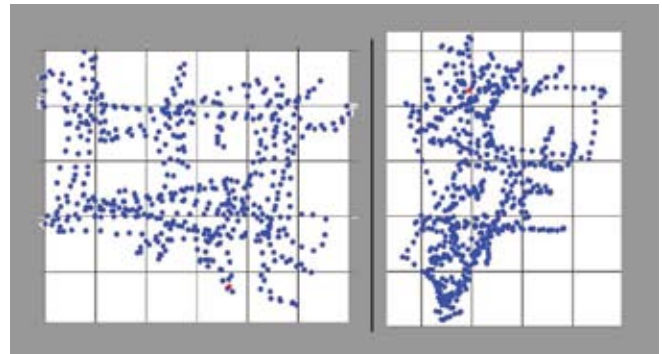


FIGURE 1 Plots from two walks around an office environment. Shown is ZUPT-aided inertial navigation based on a foot mounted IMU.

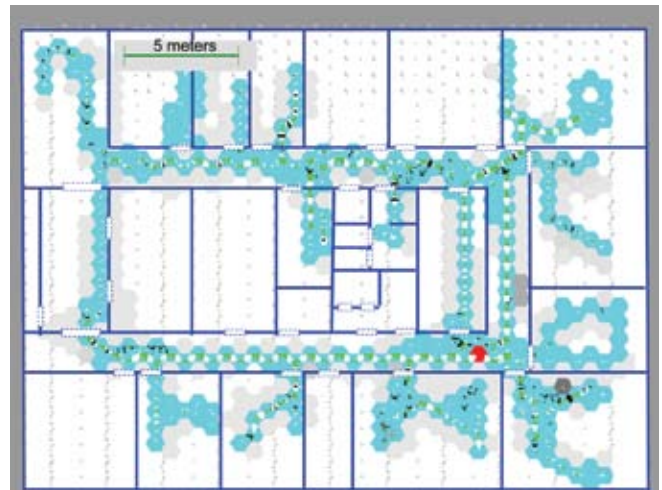


FIGURE 2 Building environment in which the data from Figure 1 was recorded; one rectangular corridor circuit and rooms to the inside and outside of it. Also shown in light blue is an overlay of a small map learnt by FootSLAM after a 13 minute walk.

(resting phase of the foot) and enables us to estimate some of the IMU sensor error states because we know that the velocity of the sensor array at that point in time must be zero in all axes.

Nevertheless, errors still accrue over time, especially the heading error, which is only weakly observable from these ZUPTs. This can be seen in the plots in **Figure 1**, which are derived from ZUPT-aided inertial navigation data logged during two walks around an office environment shown schematically in **Figure 2**.

This figure also shows an overlay of a hexagon map learnt with FootSLAM, based on a walk of about 13 minutes; in the article we will explain how these maps are learnt by the system. The FootSLAM mapping process learns which areas are accessible to a pedestrian and our short walk used to generate the map in Figure 2 only covered the corridor and some of the rooms. The more an area is explored, the more complete the resulting map will become.

Aiding can be performed using a magnetometer but this sensor also suffers from deviations in the observed magnetic field, especially indoors. Of course, aiding with satellite navigation systems (e.g., GPS) when available allows for reasonable accuracy outdoors and during short GPS outages.

Recently, three groups independently showed that foot mounted indoor positioning systems work remarkably well when aided by known building layouts or maps. (See the articles in Additional Resources by O. Woodman and R. Harle, S. Beauregard et alia, and B. Krach and P. Robertson.) Because we can reasonably assume that pedestrians are not able to walk through walls, such information should be used by any optimal or close-to-optimal positioning system.

In the cited work, researchers used particle-filtering algorithms to incorporate the map information in order to constrain particle movement to within the areas accessible to a pedestrian. Particle filters (PF), also known as sequential importance sampling, are a member of a large family of algorithms that are more or less optimal in the Bayesian filtering sense.

By incorporating a sufficiently accurate map that constrains pedestrian movements adequately, we can achieve long-term error stability can be achieved when the map is sufficiently accurate and sufficiently constrains the motion. Most indoor environments, such as offices and public buildings, allow us to meet both criteria.

These maps can be derived from sensor data collected by individuals who themselves have no need for positioning, the data being processed offline.

The use of maps is also quite natural for indoor LBS applications such as being directed towards a particular office, because a given geographic coordinate would usually have to be placed in the context of a symbolic location, such as a room number or corridor section within a building. In order for this approach to work, the map information needs to be known and free of major inaccuracies.

Robotic SLAM

For many years the robotics community has used various sensors, such as laser ranging scanners and cameras, to perform high-precision positioning of robots in buildings. Nearly two decades ago, researchers from this community introduced SLAM as a way of allowing robots to navigate in *a priori* unknown environments. (See the article by R. Smith et alia in Additional Resources.)

In robotic SLAM, a moving robot explores its environment and uses on-board sensor information and odometry control inputs to build a “map” of landmarks or features. Odometry usually refers to the control signals given to the driving wheels of the robot — and we can characterize the simple integration of these odometry signals as a form of dead reckoning.

SLAM methods fall into two main categories: EKF-SLAM, which employs an extended Kalman filter (EKF) to represent the large joint state space of robot pose (position and orientation) and all landmarks identified up to a given point in time, and FastSLAM, which uses a Rao-Blackwellized particle filter in which each particle effectively represents a pose and set of

independent compact EKFs for each landmark. (See discussion in the article by M. Montemerlo et alia cited in Additional Resources.)

The conditioning on a pose allows the landmarks to be estimated independently, thus leading to lower complexity. SLAM implementations for robot positioning always build on sensors and robot odometry, as these are readily available on robot platforms. The sensors can, for example, consist of laser rangefinders or a single or multiple cameras mounted on the robot platform. Landmark features are extracted from the raw sensor data.

Simultaneous localization and mapping is considered to be a “difficult” problem, in contrast to the two easier special cases: positioning in an environment with known landmarks or building a map of features given the true pose of the robot or other platform.

SLAM for Pedestrian Dead-Reckoning

This article builds on both areas of prior work on pedestrian positioning using foot mounted IMUs as well as the just-described SLAM approach used in robotics. Our application is human pedestrian positioning based on SLAM — that is, the difficult case where no map is available beforehand.

The main difference from robotic SLAM is that our method uses no visual or similar sensors at all. In fact, the only sensors used are the foot-mounted IMU and, optionally, a magnetometer and GPS receiver. In this article, we show that a pedestrian’s location and the building layout can be jointly estimated by using the pedestrian’s odometry alone, as measured by the foot-mounted IMU.

We have confirmed our approach by using real data obtained from a pedestrian walking in an actual indoor environment. Our experiments involved no simulations, and we will present the results from these in later sections.

Our application domain differs significantly from that of traditional uses of robotic SLAM by primarily seeking to achieve automated generation of maps that can be used later by people wishing to navigate in the building. These maps can be derived from sensor data collected by individuals who themselves have no need for positioning, the data being processed offline.

Employing such user-generated data significantly reduces the computational requirements (because processing need not be real-time), and our approach still works even in cases where the joint estimation of position and maps would yield a very large position uncertainty during the time when a building is first visited.

Although our work indicates that an accurate position estimate can be maintained after (and often prior to) convergence of the algorithm for the foot-mounted IMU upon “loop closure” (when the pedestrian retraces previous steps or areas), this is not a necessary condition in our application. Better sensors might, however, make the achievement of real-time FootSLAM very practical.

Before moving on to the theoretical basis, we shall present a brief intuitive explanation of the FootSLAM algorithm.

As pointed out earlier the odometry errors mainly affect the pedestrian's estimated heading. Consequently, the measured track will be distorted much in the same way as randomly bending a piece of wire that originally represented the track accurately.

FootSLAM works by testing a very large number of randomly chosen correcting adjustments to the distorted piece of wire and rewarding those adjustments that lead to a regular structure. The algorithm also imposes a cost in that large adjustments are more expensive than smaller ones; this avoids over-fitting.

In fact, this balance of map regularity and adjustment penalty is the direct result of the optimal mathematical FootSLAM derivation. The algorithm we use to achieve this belongs to the family of "Sequential Monte Carlo" or "particle filtering" techniques.

Theoretical Basis of FootSLAM

Human motion is a complex stochastic process, and we need to model it in a sufficiently simple fashion in order to develop our FootSLAM model and the algorithms that build on this model.

A person may walk in a random fashion whilst talking on a mobile phone, or they might be following a more or less directed trajectory towards a certain destination. Such phases of motion are governed by the person's inner mental state and, consequently, cannot be easily estimated.

As described in the paper by M. Khider et alia, a two-state Markov process can be used that allows a model of human motion to oscillate between a more random motion and targeted motion.

In order to understand the concept of the kind of map that we generate and use in our work, consider the following situation: An able-sighted person (for our purposes, equipped with FootSLAM) is standing inside a shopping center facing a wide area in front of him (see **Figure 3**). The next step(s) that this person chooses to take is influenced by two main kinds of governing factors:

- the presence of nearby *physical constraints*, such as walls, obstacles, other people, and so forth
- the presence of *visual cues* in the environment that allow the person to orientate himself and enable him to decide which future trajectory he wishes to follow in order to achieve some higher level goal, such as reaching a destination.

In contrast to robotic SLAM we have, of course, no direct access to the visual cues that our subject sees. However, we do have noisy measurements of the resulting motion, i.e., the steps taken from an initial position. In a way we can state that we *implicitly* observe these physical constraints and visual features/cues by observing the pedestrian's steps as measured by the foot-mounted IMU.

The subject may now wish to enter the wide area in front or may actually be on the way down to the level below. Knowledge of previous motion would allow us to infer more about possible future actions.



FIGURE 3 Illustration of some of the visual cues that individuals might use to orient themselves and to plan a trajectory. Shown are some obstacles in the direct vicinity as well as some elements that serve as landmarks (ovals).

In principle we could take one of two approaches: either interpret the scene and somehow infer from the overall context what steps are most likely to follow next, or observe many previous similar trajectories by the same person or other people in this environment and make a prediction based on such a learnt Markov process.

For our work we follow the second approach and limit the associated Markov process to just a single step (first order). In other words, we will represent the possible future next step of the subject based only on their current location, and we will learn the probabilities of each possible next step through observation. This would be simple enough if we had perfect knowledge of the person's location at all times, just as robotic map learning is simple in the case of a known pose.

In a nutshell, we will follow the FastSLAM approach whereby each particle assumes a certain pose history and estimates the motion probabilities conditioned on its particular assumption. Given a sufficient number of particles we can, in principle, cover the entire space of possible position histories.

Particles are weighted by how "compatible" their motion is with their previous observations of how the subject had walked when in a certain position. As we shall see, the algorithm converges remarkably quickly as long as the person revisits locations once or twice during the estimation process.

Our Model as a Dynamic Bayesian Network

Our work is based on a theoretically well-grounded representation of the Dynamic Bayesian Network (DBN) that represents the pedestrian's location, her past and present motion, the step measurements computed by the lower level EKF, and the "map" (see **Figure 4**).

This approach is used in all kinds of sequential filtering problems where noisy observations are used to estimate an evolving sequence of hidden states. Each node in the DBN represents a random variable and carries a time index. Arrows from one state variable to the next denote causation (in our

interpretation); so, arrows can never go backwards in time.

The arrows can be read in this way: all incoming arrows originate in state variables (parent states) that influence the value — in a probabilistic sense — of the target (child). In this DBN we have the following nodes (random variables):

- *pose* \mathbf{P}_k : The location and the orientation of the person in two dimensions (2D) (with respect to the main body axis)
- *step vector* \mathbf{U}_k : the change from pose at time $k-1$ to pose at time k . See **Figure 5**. Bear in mind that the step transition vector \mathbf{U}_k has a special property: knowing the old pose \mathbf{P}_{k-1} and the new pose \mathbf{P}_k enables us to determine the step transition \mathbf{U}_k entirely — just as knowledge of any two of the states \mathbf{P}_{k-1} , \mathbf{P}_k , and \mathbf{U}_k determines the unknown variable.
- *inertial sensor errors* \mathbf{E}_k : all the correlated errors of the inertial system, for instance, angular offsets or drifts.
- *step measurement* \mathbf{Z}_k^U : A measurement subject to correlated errors \mathbf{E}_k as well as white noise. See **Figure 6** for a detailed definition of the pertinent coordinate systems and step representations. Note that $p(\mathbf{Z}_k^U | \mathbf{U}_k, \mathbf{E}_k)$ encodes the probability distribution of the step measurement conditioned on the true step vector and the inertial sensor errors.
- the *visual cues* which the person sees at time k : \mathbf{Vis}_k .
- the *intention* of the person at time k : \mathbf{Int}_k is memoryless in that the resulting intention given a visual input is fully encoded in the probability density $p(\mathbf{Int}_k | \mathbf{Vis}_k)$.
- the *map* \mathbf{M} is time invariant and can include any features and information (such as human-readable signs) to let the pedestrian choose \mathbf{Int} .

Our overall goal is to estimate the states and state histories of the DBN given the series of all observations $\mathbf{Z}_{1:k}^U$ from the foot-mounted IMU (and any additional sensors, if they are present). The goal in a Bayesian formulation is to compute the joint posterior,

$$p(\mathbf{P}_{0:k}, \mathbf{U}_{0:k}, \mathbf{E}_{0:k}, \mathbf{M} | \mathbf{Z}_{1:k}^U) = p(\{\mathbf{P U E}\}_{0:k}, \mathbf{M} | \mathbf{Z}_{1:k}^U) \quad (1)$$

which following the RBPF particle filtering approach we can factorize into

$$p(\mathbf{M} | \{\mathbf{P U E}\}_{0:k}, \mathbf{Z}_{1:k}^U) \cdot p(\{\mathbf{P U E}\}_{0:k}, \mathbf{Z}_{1:k}^U) = p(\mathbf{M} | \mathbf{P}_{0:k}) \cdot p(\{\mathbf{P U E}\}_{0:k} | \mathbf{Z}_{1:k}^U) \quad (2)$$

We need to emphasize that the additional states — encoding vision and intention — of our pedestrian are never actually used; they only serve as important *structural* constraints in the DBN (linking \mathbf{P}_{k-1} and \mathbf{M} as “parent” nodes of \mathbf{U}_k). The further steps of the formal derivation of the Bayesian Filter and the RBPF particle filter are described in the article by P. Robertson et alia listed in Additional Resources.

Pedestrian Steps and Step Measurements

In this section we will show details on how we represent the step transition vector between two steps that a person takes (see Figure 5) and also discussed in further detail in the article by B. Krach and P. Robertson (Additional Resources).

In order to separate the process of updating the inertial computer driven by the IMU and the ZUPTs from the overall SLAM estimation, we have resorted to a two-tier processing in which a low-level extended Kalman filter computes the length and direction change of individual steps. This step estimate is then incorporated into the upper level particle filter in the form of a measurement. Note that this is a math-

ematical model that links the measurements received from the lower level EKF to the modeled pedestrian and his/her movement, as well as a simple representation of errors that affect the measured step.

We define a *step* to be the movement of the shoe that is equipped with the IMU from one resting phase to the next. The transition and orientation change of the foot is strongly coupled to that of the rest of the body: we assume the position of the pedestrian to be that of the relevant foot but will follow a different definition of the person’s orientation.

The orientation of the pedestrian could be interpreted as where the person is looking (head orientation). In our context, however, we find it more useful to interpret the main body axis as defining orientation, because this axis is usually close to that of the foot.

We then introduce an angular deviation between this body orientation in

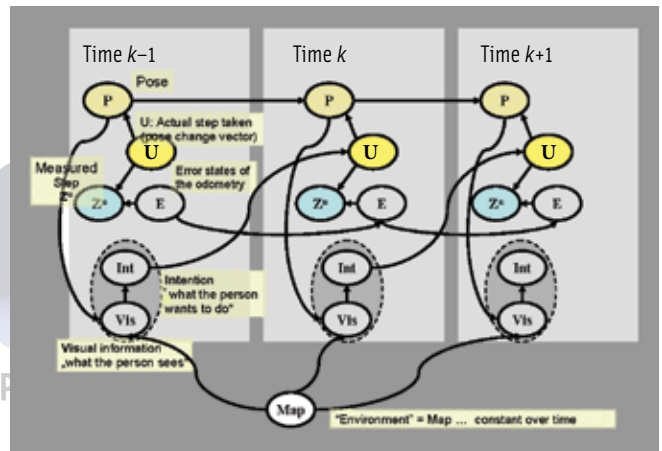


FIGURE 4 Dynamic Bayesian Network (DBN) for FootSLAM showing three time slices and all involved state (random) variables. The *map* can include any features and information to let the pedestrian choose their Intention *Int*. This DBN is the basis for the formal derivation of the Bayesian filtering algorithm.

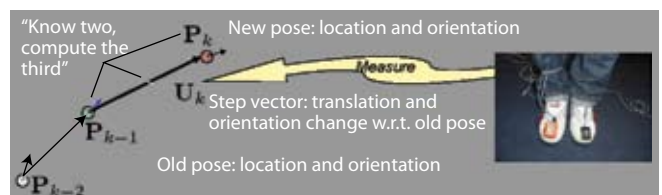


FIGURE 5 Definition of the step and its measurement. We use the notation \mathbf{U} to denote similarity to robotic odometry. In humans, the true pose change \mathbf{U} is always unknown – it is the actual step taken. In robotics \mathbf{U} is the known control input to the motors. The pertinent coordinate systems and error sources are explained in Figure 6.

space and that of the foot (IMU). This interpretation is important when we draw on additional body-mounted orientation sensors such as a magnetometer.

The complete system has, in total, four coordinate systems:

- the IMU local reference system with respect to the beginning of step measurements (i.e., INS calculation) at the lower filtering level
- a coordinate system aligned to the heading of the IMU at the last step rest phase at the lower filtering level (called *IMU zero heading*)
- a coordinate system at the higher level filter aligned to the heading of the person's body at the last step rest phase (called *person zero heading*)
- the global navigation coordinate system at the higher level filter in which the position estimate and orientation are computed (as well as the map).

In **Figure 6** we have shown the final three of these coordinate systems but have not explicitly represented the angles linking the last two (which are trivial). We assume that the step measurement suffers from both additive white translational noise and white noise on the estimated heading change of the IMU.

Moreover, we assume that an additive colored angular error appears between the true directional change of the person's body and that measured by the INS (which we call *odometry heading drift*). Lastly, we assume a very slowly changing angular offset between the person's body heading and IMU heading — for illustrative purposes we call this the *duck angle* because such an animal would typically experience a large static angular deviation if equipped with an IMU mounted on its outward-pointing foot.

Because we assume that the additive noise components are white, they do not form a part of the error state of the IMU. We do, however, model the “duck angle” as well as the odometry heading drift as random walk processes and they are formally encoded in the state variable E_k . Hereby we allow the IMU heading drift to be unbounded but restrict the “duck angle” random walk process to ± 20 degrees (essentially limited by human physiology).

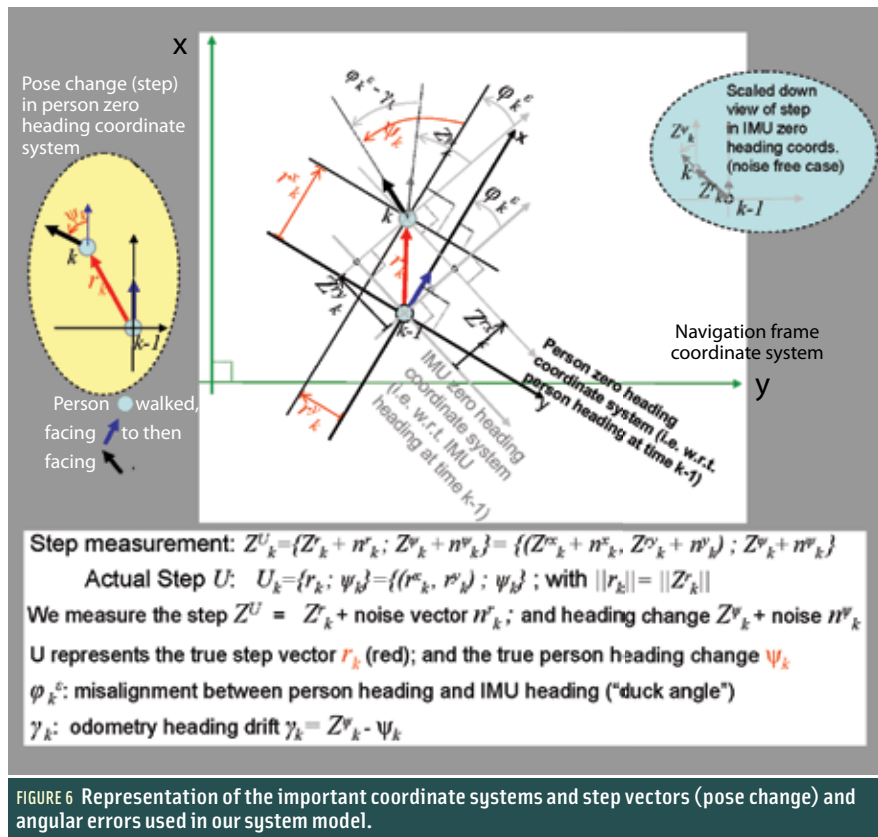


FIGURE 6 Representation of the important coordinate systems and step vectors (pose change) and angular errors used in our system model.

Map Representation in Practice

As discussed earlier, in our model the map is a probabilistic representation of possible human motion based on the subject's location in a certain part of a building. It can be interpreted in this way: a person's next step will be determined only by his or her current location in the sense that each future step is drawn from a location-dependent probability distribution. This corresponds to the fictive pedestrian behavior in which a person looks at a probability distribution posted at each location, and “draws” the next step using exactly this governing distribution.

As mentioned previously, we resort to a Rao-Blackwellized particle filter (RBPF) that follows a FastSLAM partitioning in which each particle represents the pedestrian's location track and a probabilistic representation of possible motion for each location in a two-dimensional space. This means that we are representing human motion as a first-order Markov process: the next step taken by the pedestrian is solely a proba-

bilistic function of the current location.

In order to compute and store the probability distribution of human motion as a function of location we have chosen to partition the space (restricted so far to two dimensions) into a regular grid of adjacent and uniform hexagons of a given radius (e.g., 0.5 meters, see **Figure 7**).

Every particle holds (conditional) estimates of the transition probabilities across the edges of all visited hexagons and updates these based on the motion taken by the particle hypothesis. We assume a uniform prior in our Bayesian estimation of these probability distributions and will return to this point in later discussion. Furthermore, a particle explores possible deviations of the true pedestrian's path as a result of the sequence of the IMU errors $E_{0:k}$ (refer to **Figure 8**).

When we use a large number of particles (N_p) in the particle filter, we are exploring a very large space of possible IMU errors and at least one of them will usually be very close to the true IMU error sequence. The larger we make N_p

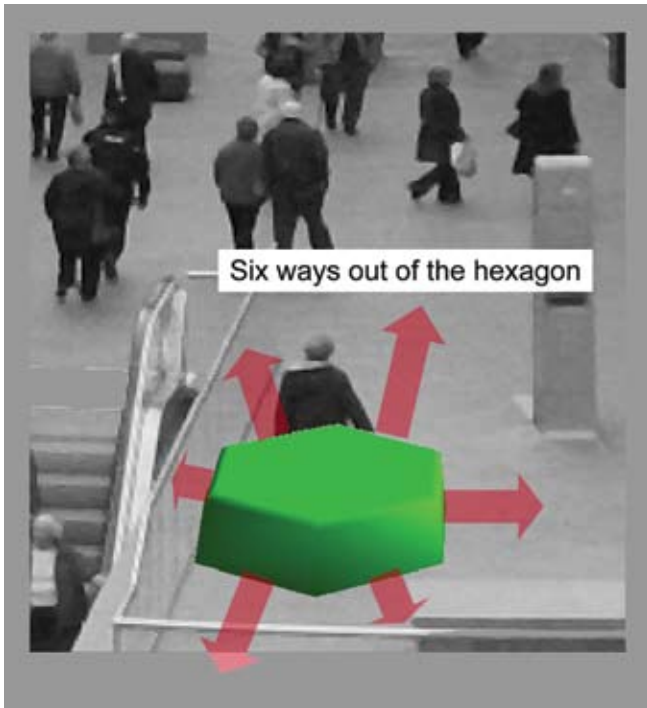


FIGURE 7 2D hexagon representation for stochastic pedestrian movement used in this article. We represent the six probabilities for crossing each of the hexagons in a regular grid with adjacent hexagons.

the more reliable FootSLAM becomes.

A particle's weight is updated according to the previously estimated knowledge of the state transitions for the outgoing edges of the hexagon it is cur-

rently occupying. As a result, particles will be "rewarded" (given additional weight) by revisiting similar transitions in space. Let us now specify the probabilistic map formally, based on transitions across the hexagon grid. We assume a two-dimensional position domain and populate space with adjacent hexagons of radius r . We can restrict this space to the region visited by any particle and define $H = \{H_0, H_1, \dots, H_h, \dots, H_{N_H-1}\}$ as the set of N_H hexagons, where the index h uniquely references a hexagon's position. Furthermore, we define $M_h = \{M_h^0, M_h^1, \dots, M_h^5\}$ as the set of transition probabilities across the edges of the h -th hexagon and

rently occupying. As a result, particles will be "rewarded" (given additional weight) by revisiting similar transitions in space.

Let us now specify the probabilistic map formally, based on transitions across the hexagon grid. We assume a two-dimensional position domain and populate space with adjacent hexagons of radius r . We can restrict this space to the region visited by any particle and define $H = \{H_0, H_1, \dots, H_h, \dots, H_{N_H-1}\}$ as the set of N_H hexagons, where the index h uniquely references a hexagon's position. Furthermore, we define $M_h = \{M_h^0, M_h^1, \dots, M_h^5\}$ as the set of transition probabilities across the edges of the h -th hexagon and

$$M_h^e(U_k) = P(\mathbf{P}_k \in H_j | \mathbf{P}_{k-1} \in H_h) \quad (3)$$

s.t. H_j is reached from H_h by U_k ,

Here, $j \neq h$ — that is, we moved to a new hexagon, where $0 \leq e(U_k) \leq 5$ is the edge of the outgoing hexagon associated with U_k , i.e., the edge of the hexagon in which \mathbf{P}_{k-1} lies and that borders the hexagon in which \mathbf{P}_k lies. (See Figure 9.)

Also, we can state that $\sum_e M_h^e = 1$. When M_h^e is written in boldface we are denoting a random variable. We thereby introduce the notion that M_h^e , a probability, is unknown to us. We only have estimates of $p(M_h^e | \mathbf{P}_{0:k-1})$ that are the result of observations of the sequence of positions up to step k .

Our map random variable \mathbf{M} is defined as the set:

$$\mathbf{M} = \{\mathbf{M}_0, \mathbf{M}_1, \dots, \mathbf{M}_h, \dots, \mathbf{M}_{N_H-1}\}, \quad (4)$$

where \mathbf{M}_h is a random variable vector of length 6 denoting the transition probabilities of the hexagon with index h . In the following we will write \hat{h} for outgoing hexagon $h(\mathbf{P}_{k-1})$, and \bar{e} for the crossed edge $e(U_k)$ for brevity.

Learning the transition map on a particle-by-particle basis is very easy and is based on Bayesian inference of multinomial and binomial distributions. Each time a specific particle with index i makes a transition $\mathbf{P}_{k-1}^i \rightarrow \mathbf{P}_k^i$ across hexagon edge \bar{e} we count this transition in the local map of hexagon $H_{\hat{h}}$ for particle i .

Summary of the RBPF Algorithm

We can take good advantage of a *Likelihood Particle Filter* as is described in the article by S. Arulampalam et alia because the step measurement \mathbf{Z}_k^U is

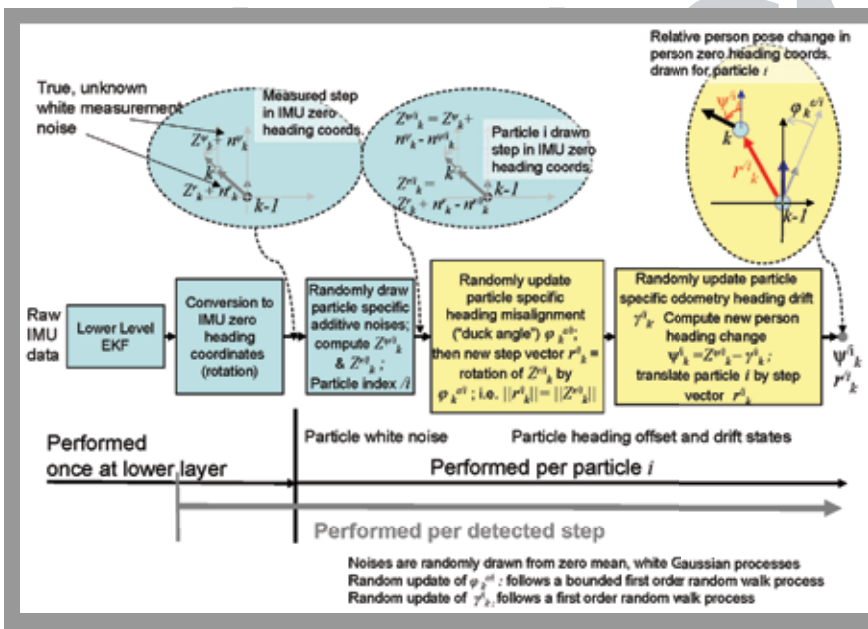


FIGURE 8 Processing chain from step estimates at the lower filtering level up to the stage where particles are drawn from the proposal density Equation (5). Drawing the two odometry error state angles from their corresponding random walk processes corresponds to drawing \mathbf{E}_k^i from $p(\mathbf{E}_k^i | \mathbf{E}_{k-1}^i)$. Drawing the two white noise processes for $n^{w/i}$ and $n^{h/i}$, and then applying the new angles stored in state \mathbf{E}_k^i , results in drawing the new step vector \mathbf{U}_k^i from $p(\mathbf{U}_k^i | \mathbf{Z}_k^U, \mathbf{E}_k^i)$ — as defined within the person zero-heading coordinates.

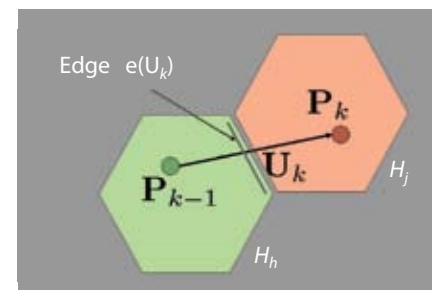


FIGURE 9 Definition of the hexagon transition $M_h^e(U_k)$ in Equation (3)

very accurate. Weighting with a “sharp” likelihood function $p(\mathbf{Z}_k^U | \mathbf{U}_k, \mathbf{E}_k)$ would cause most particles outside the measurement to receive very low weight and effectively be wasted. Thus, we sample using the likelihood function rather than sampling from the state transition model.

Specifically, we have chosen the proposal density of the particle filter to be

$$q(\{\mathbf{P}^U \mathbf{E}\}_k | \{\mathbf{P}^U \mathbf{E}\}_{0:k-1}, \mathbf{Z}_{1:k}^U) \hat{=} \quad (5)$$

$$p(\mathbf{E}_k | \mathbf{E}_{k-1}) \cdot p(\mathbf{U}_k | \mathbf{Z}_k^U, \mathbf{E}_k).$$

We will discuss the practical issues involved in a real implementation of our RBPF algorithm in a moment, but first we will describe its general operation:

1. Initialize all N_p particles to $\mathbf{P}_0^i = (x=0, y=0, h=0)$ where x, y, h denote the pose location and heading in two dimensions; draw \mathbf{E}_0^i from a suitable initial distribution for the error states.

2. For each time step increment k :

(a) Given the latest step measurement \mathbf{Z}_k^U , particles with index i are drawn from the proposal density $p(\mathbf{E}_k | \mathbf{E}_{k-1}) \cdot p(\mathbf{U}_k | \mathbf{Z}_k^U, \mathbf{E}_k^i)$ (see **Figure 8**).

(b) The pose \mathbf{P}_k^i is computed by adding the vector \mathbf{U}_k^i to \mathbf{P}_{k-1}^i ; also updating the heading of the pedestrian according to \mathbf{U}_k^i .

(c) The particle weight updates are simply

$$w^i \propto w_{k-1}^i \cdot \left\{ \frac{N_{\tilde{h}}^e + \alpha_{\tilde{h}}^e}{N_{\tilde{h}} + \alpha_{\tilde{h}}} \right\}^i \quad (6)$$

where the counts are those that are computed up to step $k-1$: The term $N_{\tilde{h}}^e$ is the number of times particle i crossed the transition; $N_{\tilde{h}}$ is the sum of the counts over all edges of the hexagon in this particle’s map counters. The terms $\alpha_{\tilde{h}}^e$ and $\alpha_{\tilde{h}} = \sum_{e=0}^5 \alpha_{\tilde{h}}^e$ are the priors of this map segment. (In our experiments we chose $\alpha_{\tilde{h}}^e = 0.8$ for all edges and hexagons.)

(d) Particle weights are normalized to sum to unity.

(e) Recompute $\{N_{\tilde{h}}^e\}^i$ for the transition from \tilde{h}^i s.t. $\mathbf{P}_{k-1}^i \in H_{\tilde{h}^i}$, and the transition \tilde{e}^i corresponds to the step ending at \mathbf{P}_k^i . The counts are kept for each particle and hence store the entire history of that particle’s path

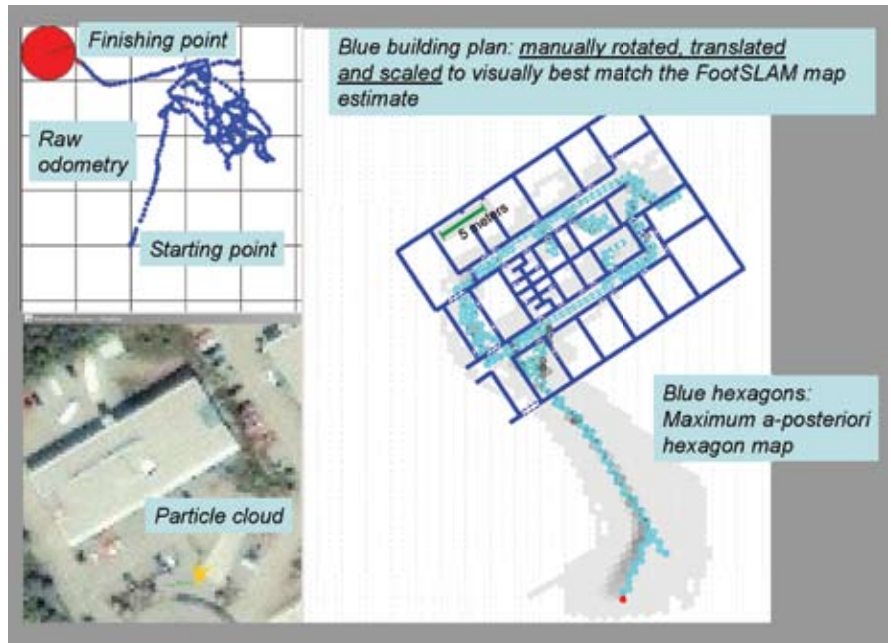


FIGURE 10 Result of FootSLAM processing for an outdoor-indoor-outdoor walk. Top left: raw odometry, where we see how the starting and finishing point are far apart (in reality, they were very close together). Right: resulting FootSLAM map at the end of the processing run through the entire data set. Bottom left: particle cloud (note: smaller scale compared to the map window shown on right). In this experiment we assumed that we knew the location of the building’s outer walls but not the entrance doors or the interior layout.

through the hexagon grid. They are used in Equation (6) the next time $H_{\tilde{h}^i}$ is visited by this particle.

f) Resampling can be performed if required.

A number of implementation issues need to be addressed in order for the algorithm to work in practice. First of all, when computing the counts for each particle, we in fact assume that observing a certain transition from an outgoing hexagon to an incoming one allows us to increment the counts for the outgoing as well as the incoming hexagon (on the appropriate edge). This is the same as assuming that a person is equally likely to walk in either direction and that we should not waste this information.

Next, we have assumed so far that an increment of the time index k is associated with a step that leads from one hexagon to an adjacent one. In reality a step might keep us in the hexagon or it might lead us over several hexagons.

To address this we simply perform a weight update only when we have stepped out of the last hexagon and apply multiple products in the weight update (6) for all edges crossed if the step was

a larger one. Similarly, we update the counts of all edges crossed between \mathbf{P}_{k-1}^i and \mathbf{P}_k^i .

We also incorporated a small correction term in the weight update equation (6) (raising it to a power depending on the step vector angle within the hexagon grid) in order to account for the fact that straight tracks with different angles traversing the grid will each yield a slightly different total number of hexagon edge transitions per distance traveled. (Otherwise, particles with some directions would be slightly favored.)

Experiments and Data Processing

Our results based on real data are very promising. In several runs lasting up to about 12 minutes each, we collected the data of a sensor-equipped pedestrian walking in office environments as well as in the adjacent area outside.

We first present the quantitative results for the important scenario outdoors-indoors-outdoors (**Figures 10, 11, 12, and 13**). This represents the case where a person uses GPS outdoors and enters a building and leaves it again at some point.

There are two main applications for this: The true SLAM scenario where we wish to locate the user in real time while they are walking, and the map-building scenario where maps are used later by other people.

The recorded sensor data is collected during the walk and is then processed offline in our RBPF implementation. In our visualizations, the location estimate (RBPF 2D particle cloud) and the current *maximum a posteriori* estimate of the probabilistic hexagon map are displayed during the processing of the data — as well as the raw step calculation of the lower level EKF (see, for example, Figure 10).

This allows interpretation and explanation of important effects, such as the evolving map hypotheses. A collection of our data processing runs were recorded as video and are available on the Internet at <http://www.kn-s.dlr.de/indoornav>.

In Figure 10 and Figure 11 we show qualitative results for the cases where we, respectively, assumed and did not assume a rough knowledge of the building outline. In Figure 12 we show the positioning error during the walk in comparison to other approaches.

Note that unaided FootSLAM, as with other forms of SLAM, is rotation- and translation-invariant, because there is no absolute position or orientation reference. When combined with GNSS, for instance, the invariance and the resulting ambiguity is much reduced because we are using an additional anchor in the position domain, at least for a part of the time.

In our example with the outdoors-indoors-outdoors scenario we have a residual translation ambiguity of the map that is in the order of the GNSS accuracy. Sometimes a residual rotation ambiguity also appears because we have only anchored our building map to the GNSS location on one side — the entrance.

This does not create a problem for a relative localization using this map (such as finding an office) and can be improved with knowledge of the outer walls. Future work will focus on merg-

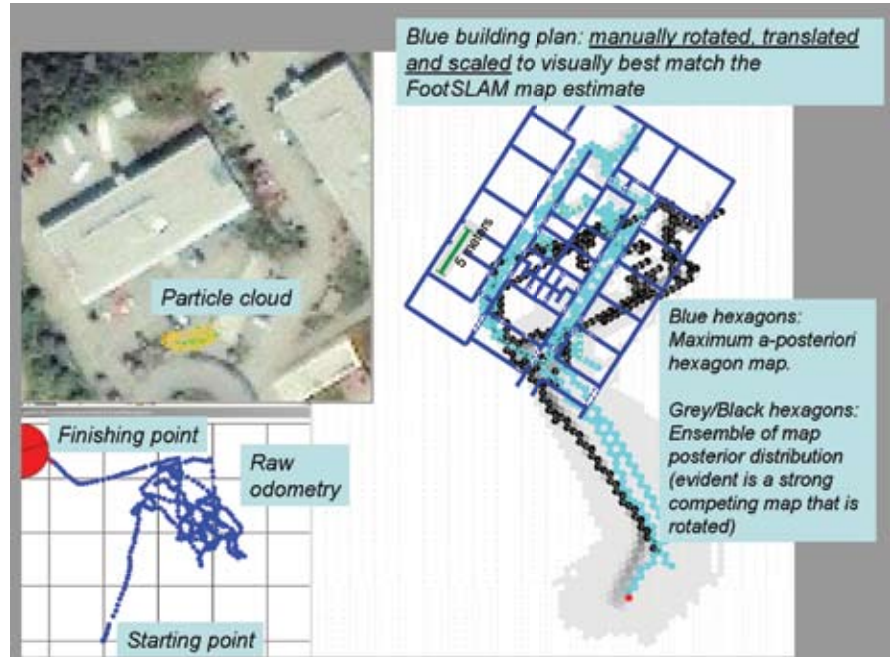


FIGURE 11 Result of FootSLAM processing for an outdoor-indoor-outdoor walk. Bottom left: raw odometry. Right: resulting FootSLAM map. Top left: particle cloud (note: smaller scale compared to the map window shown on right). In this experiment we did not assume that we knew the building's outer walls. Observe how two main hypotheses for maps survive. This is to be expected given the rotation-invariant nature of SLAM.

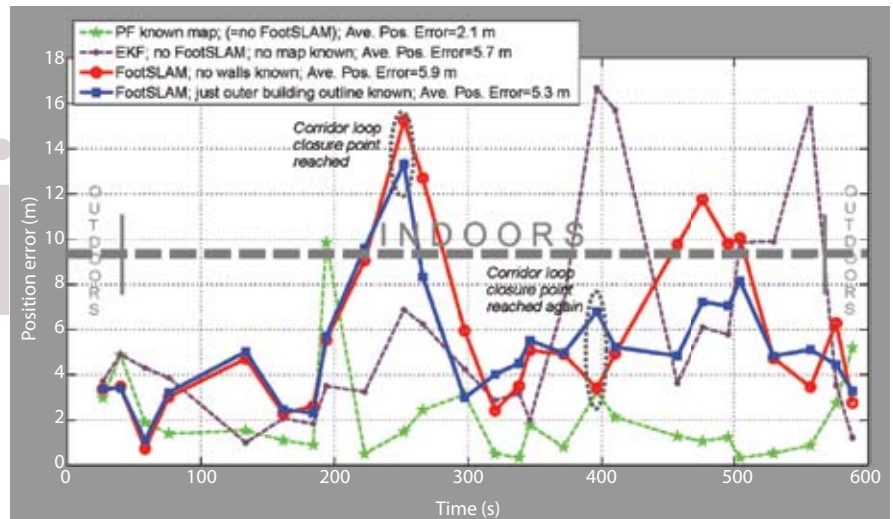


FIGURE 12 Position error of FootSLAM processing for the outdoor-indoor-outdoor walk of Figures 10 and 11 compared with particle filter using complete building plan information and a simple EKF using no such information. The FootSLAM results were averaged over three RBPF runs over the data set with 55,000 particles each. The EKF and PF curves are for a single run.

ing maps from many users; because the errors from each run are essentially uncorrelated, we expect these kinds of influences to be averaged out.

We have also processed the three-dimensional data from the outdoor-indoor-outdoor walk portrayed in the previously mentioned online video (data

set “March09Measurement03”) and discussed in the article by M. Angermann et alia.

Assuming that this had been a two-dimensional measurement, no major degradation occurs as the floor plan corridors are more or less identical/compatible at different levels of the building.

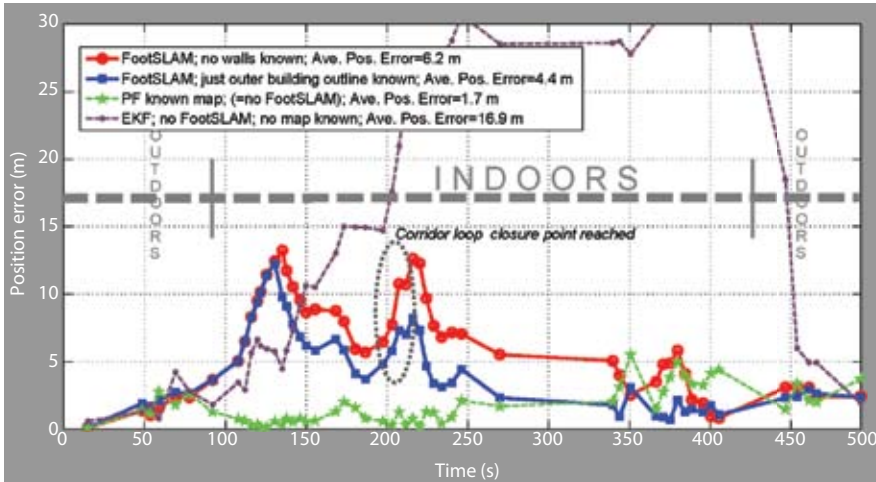


FIGURE 13 Position error of FootSLAM processing for the outdoor-indoor-outdoor walk of [10] [11]. Comparison with particle filter using complete building plan information and a simple EKF using no such information. Note: FootSLAM assumes this as being two-dimensional which in this originally 3D data set causes no major degradation as the floor plan corridors are more or less identical/compatible at different levels. The FootSLAM results are a single RBPF run each, with 35000 particles. The EKF and PF curves are also for a single run.



FIGURE 14 A further result of an indoor-only experiment in a different location. The map was generated using 6,000 particles to process 12 minutes of data in faster than real-time on a standard personal computer using a Java implementation.

Obviously we ignored the altimeter data but included the magnetometer. The results are shown in **Figure 13**.

Results In order to evaluate the first case we measured the position accuracy over time, during the entire walk. To validate the second application, we show qualitatively the resulting map, created using all the data up to the end of the walk (that is, when we are outdoors again).

In a subset of our evaluations we assumed that we knew *a priori* the location of the outside building walls to within three meters of the true wall locations. This helps the FootSLAM to converge a little, but it is not a requirement.

It is realistic, however, for somebody mapping a new region to roughly mark the outer corners of the target building manually using an online satellite image

service, for instance. The resulting coordinates can then be used to construct a simple polygon as prior map information by the particle filter. The resulting coordinates can be incorporated to construct a simple polygon as prior map information to be used by the particle filter.

In the work presented in figures 10–13, the pedestrian walked from a point outside the building, through the front door of the office, and round the corridor of the ground floor.

During the first walk round, the pedestrian entered and then left a number of rooms and then continued to walk the corridor to the next room. This pattern was repeated two more times, at which point the pedestrian exited the building by the same door.

Figure 14 shows results from an indoor-only experiment. Blue hexagons are those that are covered by the map of the best particle. The white and light grey areas to be seen within hexagons and connecting hexagons encode the frequency with which these were traversed. In this way we can easily recognize the main paths, just as we would see from an aerial view a heavily trodden path across grass.

We used 6,000 particles to process data from a run of about 12 minutes duration. The walk consisted of exploring a meeting room with a large table, a long corridor, and a canteen. Note that this experiment has no direct large loop closure, but FootSLAM also converges when the user revisits areas by backtracking without closing a large loop. Also note how the algorithm learned the location of the table to within about 1–1.5 meters — essentially limited by the natural feature dimension in the building and the discretization error due to the finite hexagon radius.

Discussion and Further Work

The true layout of the building was, of course, not used in the processing and only manually inserted over the resulting FootSLAM maps for comparison — applying rotation, scaling, and translation chosen to match each FootSLAM map. The results of the outdoor-indoor-outdoor and indoors-only trials showed

a remarkable ability of the RBPF to estimate the reachable areas. Errors of the location of the doors were usually to within about one meter and never more than about two to three meters off.

All results so far were obtained from just a single track or walk-through and assume no further processing to merge tracks. In a real system, efforts must be undertaken to resolve the scale, rotation, and translation ambiguities and errors that are often inherent in SLAM.

In our approach (where we couple with GPS in the outdoor portion and optionally a magnetometer), these ambiguities may not be so pronounced and may be locally confined to the building indoor areas. Future work should address techniques that combine maps from different sources, such as different runs from the same person or runs from different people. We believe that after a few runs the ambiguities will be averaged out.

Furthermore, the partial availability of GNSS indoors — even with large errors at any one time — will *over time* help to eliminate the ambiguities even further. In both cases the user generated approach will over time improve the quality of the maps and will also adjust to changes in the building layout.

Inspecting the numerical results, we can make the following observations:

- Observing the particle cloud during processing and also the evolution of the position error, it becomes evident that the estimator diverges at first as the area is being explored, but then begins to converge (at loop closure) closer to the true location and remains reasonably stable. The cloud naturally spreads as new areas of the building are being explored for the first time, only to converge again as the pedestrian revisits familiar ground.
- The numerical results indicate that the use of rough knowledge of the outer building walls (building perimeter) help to improve the error slightly.
- The use of perfect building plan information — not surprisingly — gives the best performance. This

is because the location of the walls is known with submeter accuracy. The result is that indoor positioning accuracy is usually better than outdoors.

- When FootSLAM is used, the accuracy cannot be better than the anchor achieved while using GPS before entering the building. This error in our experiments was typically around three to seven meters; so, this is a baseline error onto which the FootSLAM relative errors are essentially added.
- The extended Kalman filter that does not use FootSLAM diverged after some time, especially in the second data set. (Divergence is a random process and depends on the random occurrence of drifts and angular displacement of the stride estimation at the lower level and is a function of the IMU errors).

Because our maps are probabilistic, we could also estimate pedestrians' future paths — similar to work for driver intent estimation described in the paper by J. Krumm (Additional Resources). Further work should also integrate more sensors, address 3D issues, as well as collective mapping in which users collect data during their daily lives and maps are combined and improved.

Current work is addressing PlaceSLAM: the use of manually indicated markers, which are recognizable places that are flagged by the user each time she comes to this place to further aid convergence. Finally, it is important to point to new developments in sensor technology that are achieving substantial improvements to performance. (See, for example, the article by E. Foxlin and S. Wan in Additional Resources.)

This new work on sensors is important for FootSLAM. First, the more accurate sensors will allow larger areas to be mapped by FootSLAM for any given number of particles in the algorithm; so, a better sensor will allow low complexity implementation. Second, a better sensor will make it more likely that the odometry error will be small before the first FootSLAM loop closure or backtrack, meaning that real-time

FootSLAM without any form of prior map will be even more viable.

Authors



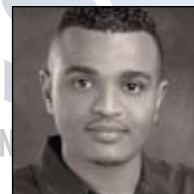
Patrick Robertson received a Ph.D. in electronic engineering from the University of the Federal Armed Forces, Munich. He is currently with DLR, where his research interests are navigation, sensor-based context-aware systems, signal processing, and novel systems and services in various mobile and ubiquitous computing contexts.



Michael Angermann received a Ph.D. in electronic engineering from the University of Ulm, Germany. He is currently with DLR, where his research interests are advanced and integrated communication and navigation systems.



Bernhard Krach received the Dipl.-Ing. degree in electrical engineering from University of Erlangen-Nuremberg, Germany. From 2005 to 2009 he was with the Institute of Communications and Navigation at DLR. Since 2009 he has been with EADS Military Air Systems.



Mohammed Khider received his masters degree in communications technology from the University Of Ulm, Germany. He obtained his B.Sc. degree in electrical engineering/communications from the UAE University, United Arab Emirates. He is currently with the German Aerospace Center and a Ph.D. student at the University of Ulm. His research interests are navigation, multi-sensor fusion, mobility models, signal processing, and context-aware services.

Acknowledgements

This research has received funding from the European Community's FP7 Program [FP7/2007-2013] under grant agreement no. 215098 of the "Persist" Collaborative Project. This article is

SLAM DANCE continued on page 64

SLAM DANCE *continued from page 58*

based in part on a paper presented at the ION GNSS 2009 conference in Savannah, Georgia, USA.

Manufacturers

An EVK-5 GPS receiver from **u-blox AG**, Thalwil, Switzerland; an OS5000 digital compass from **OceanServer Technology, Inc.**, Fall River, Massachusetts, USA; an MS55490 baro-altimeter from **MEAS Switzerland SA** (formerly **Intersema Sensoric SA**), Bevaix, Switzerland; i-CARD2 RFID reader and tags from **Identec Solutions AG**, Lustenau, Austria; an MTx-28A53G25 IMU from **Xsens Technologies B.V.**, Enschede, The Netherlands.

Additional Resources

[1] Angermann, M., and A. Friese, M. Khider, B. Krach, K. Krack, and P. Robertson, "A Reference Measurement Data Set for Multisensor Pedestrian Navigation with Accurate Ground Truth," in *Proceedings of European Navigation Conference ENC-GNSS 2009*, Naples, Italy, 2009

[2] Arulampalam, S., and S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/Non-Gaussian Bayesian Tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 174–188, February 2002

[3] Beauregard, S., Widyawan, and M. Klepal, "Indoor PDR Performance Enhancement Using Minimal Map Information and Particle Filters," in *Proceedings of the IEEE/ION PLANS 2008*, Monterey, California USA, May 2008

[4] "FootSLAM videos and reference data sets download," <<http://www.kns.dlr.de/indoornav>>

[5] Foxlin, E., "Pedestrian Tracking with Shoe-Mounted Inertial Sensors," *IEEE Computer Graphics and Applications*, vol. 25, no. 6, pp. 38–46, November 2005

[6] Foxlin, E., and S. Wan, "Improved Pedestrian Navigation Based on Drift-Reduced MEMS IMU Chip," 2010 ION International Technical Meeting, San Diego, California USA, January 2010.

[7] Khider, M., and S. Kaiser, P. Robertson, and M. Angermann, "A Novel Movement Model for Pedestrians Suitable for Personal Navigation," in *ION National Technical Meeting 2008*, San Diego, California, USA, 2008


[8] Krach B., and P. Robertson, "Cascaded Estimation Architecture for Integration of Foot-Mounted Inertial Sensors," in *Proceedings of the IEEE/ION PLANS 2008*, Monterey, California USA, May 2008

[9] Krumm, J., "A markov model for driver turn prediction," in *SAE 2008 World Congress*, Detroit, MI USA. Springer-Verlag New York, Inc., 2008

[10] Montemerlo, M., and S. Thrun, D. Koller, and B. Wegbreit, "FastSLAM: A factored solution to the simultaneous localization and mapping problem," in *Proc. AAAI National Conference on Artificial Intelligence*, Edmonton, Canada, 2002

[11] Robertson, P., and M. Angermann, and B. Krach, "Simultaneous Localization and Mapping for Pedestrians Using Only Foot-Mounted Inertial Sensors," in *Proceedings of UbiComp 2009*, Orlando, Florida, USA

[12] Smith, R., and M. Self, and P. Cheeseman, "Estimating Uncertain Spatial Relationships in Robotics," in *Autonomous Robot Vehicles*, I. J. Cox and G. T. Wilfong, Eds., Springer Verlag, New York, 1990, pp. 167–193

[13] Woodman, O., and R. Harle, "Pedestrian Localisation for Indoor Environments," in *Proceedings of the UbiComp 2008*, Seoul, South Korea, September 2008. 

Inside GNSS
GPS | GALILEO | GLONASS | COMPASS