Inertial Systems Based Joint Mapping and Positioning for Pedestrian Navigation

Patrick Robertson†, Michael Angermann†, Bernhard Krach†∗, Mohammed Khider†

† *German Aerospace Center (DLR), Institute of Communications and Navigation, PO Box 1116, D-82230 Oberpfaffenhofen, Germany. E-Mail: firstname.lastname@dlr.de*

∗ *EADS Military Air Systems, Manching, Germany*

Biography

Patrick Robertson received a Ph.D. from the University of the Federal Armed Forces, Munich, in 1995. 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. from the University of Ulm in 2004. 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, in 2005. From 2005 to 2009 he has been with the Institute of Communications and Navigation at DLR. Since 2009 he is with EADS Military Air Systems.

Mohammed Khider received his master's degree in electrical engineering from the University of Ulm, Germany. He is currently with DLR and a PhD student at the University of Ulm. His research interests are navigation, multi-sensors fusion, mobility models, signal processing and context aware services.

Abstract

We present results for a new pedestrian localisation technique that builds on the principle of Simultaneous Localization and Mapping (SLAM). Our approach is called FootSLAM since it is based mainly on the use of shoe-mounted inertial sensors that are used to measure a pedestrian's steps while walking. In contrast to SLAM used in robotics no specific feature-detection sensors such as cameras or laser scanners are needed in our approach. 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 (maps) can be learnt automatically while people walk in a building. This can be done either directly to localise this specific person or in a offline fashion in order to provide maps for other people. We have combined our system with a GPS and have undertaken experiments in the important scenario where a person enters a building from outside and walks around within this building without GPS availability. Our experiments were undertaken by recording the raw sensor data and ground truth reference information. Offline procesing and comparison with the ground truth reference information allows quantitative evaluation of the achieved localisation accuracy.

1 Introduction

Recent work has shown remarkable advances in the area of pedestrian indoor positioning aided by low cost MEMS inertial sensors. At the present time, full autonomous inertial navigation is still far from the realm of possibilities - due to sensor error induced drift which causes position errors to grow unbounded within a few seconds. The work of Foxlin on foot mounted Inertial Measurement Units (IMUs) has shown how zero velocity updates - ZUPTs- during the rest phase of a pedestrian's foot can be used to solve the problem of non-linear error growth over time [1]. This is because the inertial navigation system (INS) is able 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 (resting phase of the foot) and allows us to estimate some of the IMU sensor error states since we know that the velocity of the sensor array must be zero in all axes. Nevertheless, errors still accrue over time, especially the heading error which is only weakly observable from these zero velocity updates (see Figure 1). 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 for 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 [2, 3, 4]. This is because it can reasonably be assumed that pedestrians cannot walk through walls and this information should be used by any optimal or close-to-optimal positioning system. In this work, the 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 which are more or less optimal in the Bayesian filtering sense. As a result of incorporating maps, long term error stability can be achieved when the map is sufficiently accurate and sufficiently constrains the motion - both these criteria are usually met in most indoor environments such as offices and public buildings. The use of maps is also quite natural, since usually any geographic coordinate would anyhow have to be placed in the context of a symbolic location, such as a room number or corridor section within a building, in order to be used for various location based services such as being directed towards a particular office. In order for this approach to work, the map information needs to be known and be free of major inaccuracies.

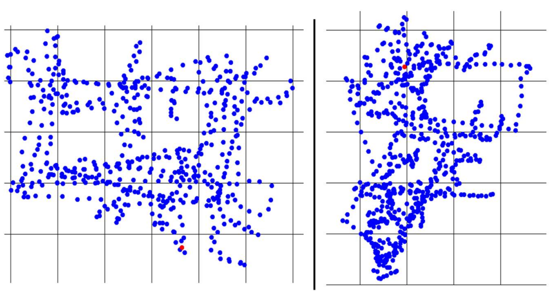

Figure 1. Plots from two walks around an office environment; see Figure 2. Shown is ZUPT aided inertial navigation based on a foot mounted IMU.

1.1 Robotic Simultaneous Localization and Mapping (SLAM)

The robotics community has for many years used numerous sensors such as laser ranging scanners and cameras to perform high precision positioning of robots in buildings. A difficult problem known as SLAM - Simultaneous Localization and Mapping - has been defined as a way of allowing robots to navigate in a-priori unknown environments [5]. In SLAM, a moving robot explores its environment

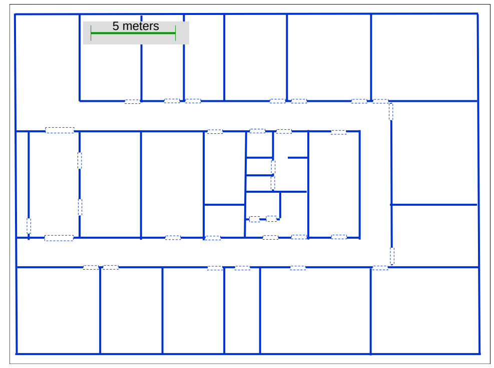

Figure 2. Building environment in which the data from Figure 1 was recorded; one rectangular corridor circuit and rooms on the inside and outside of it.

and uses its 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 simple integration of these odometry signals can be seen as a form of dead reckoning. There are two main categories of SLAM: EKF-SLAM that employs an extended Kalman Filter (EKF) to represent the large joint state space of robot pose (position and orientation) and all landmarks identified so far. The approach known as FastSLAM uses a Rao-Blackwellized Particle Filter (RBPF) where each particle effectively represents a pose and set of independent compact EKFs for each landmark [6]. 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 rangers or a single or multiple cameras mounted on the robot platform, and the features are extracted from the raw sensor data. Simultaneous Localization and Mapping is considered to be a “hard” 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.

1.2 SLAM for pedestrian dead-reckoning

This paper will build on both areas of prior work on pedestrian positioning using foot mounted IMUs as well as the SLAM approach used in robotics which has been described above. Our application is human pedestrian positioning based on SLAM - i.e. the difficult case where no map is available a-priori. However, the main difference to robotic SLAM is that no visual sensors are used at all. In fact, the only sensors used are the foot mounted IMU and optionally a magnetometer and / or a satellite navigation receiver. In this paper 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 environment; no simulations were used - we will present these results in later sections.

There is another major difference between our application domain and that of the usual applications of robotic SLAM. Our goal is to primarily allow automated generation of maps that can later be used by people wishing to navigate in that building, say. These maps can be generated by sensor data collected by people who themselves have no need for positioning, the data being processed in an offline fashion. This significantly reduces the computational requirements (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. While our work indicates that an accurate position estimate can be maintained after (and often prior to) convergence of the algorithm upon "loop-closure", this is not a necessary condition in our application. Better sensors might, however, make the application of real-time FootSLAM very worthwhile.

2 Theoretical Basis

2.1 Intuitive Discussion of factors governing human pedestrian motion

Human motion is a complex stochastic process which we need to model in a sufficiently simple fashion in order to develop our FootSLAM model and the algorithms which build on it. 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 cannot be easily estimated. In [7] a two-state Markov process was used to allow a model of human motion to oscillate between a more random motion and targeted motion.

In order to understand the concept of the kind map which we will use in this work, consider the following situation: An able sighted person is standing in a shopping centre facing a wide area in front. The next step(s) which 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, etc.
- The presence of *visual cues* in the environment which allow the person to orientate themselves and to allow them to decide which future trajectory they wish to follow in order to achieve some kind of higher level goal, such as reaching a destination (see Figure 3).

Figure 3. Illustration of some of the visual cues that a person might use to orientate themselves and to plan a trajectory. Shown are some obstacles in the direct vicinity as well as some landmarks (ovals).

In contrast to robotic SLAM we of course have no direct access to the visual cues that our subject sees; we do, however, have noisy measurements of the resulting motion, i.e. the steps taken. 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 and in principle two approaches could be taken: Either interpret the scene and somehow infer from the overall context what next steps are most likely to follow, 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. In our work we will 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 for the case of 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.

2.2 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

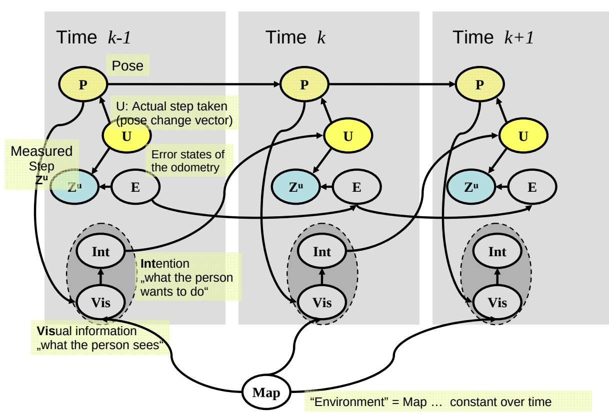

Figure 4. Dynamic Bayesian Network (DBN) for FootSLAM showing three time slices and all involved state (random) variables. The Map can include any features and information to let the pedestrian choose their **Int**. This DBN is the basis for the formal derivation of the Bayesian filtering algorithm.

the following nodes (random variables):

- Pose $\mathbf{P}_k$: The location and the orientation of the person in 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. It is important to bear in mind that the step transition vector $\mathbf{U}_k$ has a special property: given the old pose $\mathbf{P}_{k-1}$ and the new

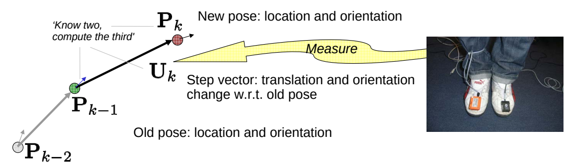

Figure 5. Definition of the step and it's measurement. We use the notation **U** to denote similarity to robotic odometry. In humans, the true pose change **U** is always unknown - it is the actual step taken. In robotics **U** is the known control input to the motors. The pertinent coordinate systems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

pose $\mathbf{P}_k$ then this determines 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 one.

- Inertial sensor errors $\mathbf{E}_k$: All the correlated errors of the inertial system. For instance angular offsets or drifts.
- Step measurement $\mathbf{Z}^U_k$: A measurement subject to correlated errors $\mathbf{E}_k$ as well as white noise. See Figure 6 for a definition of the pertinent coordinate systems and step representations. Note that $p(\mathbf{Z}^U_k|\mathbf{U}_k,\mathbf{E}_k)$ 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 **Int**.

Our overall goal is to estimate the states and state histories of the DBN given the series of all observations $\mathbf{Z}^U_{1:k}$ 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 [8],

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

which following the RBPF particle filtering approach we can factorize into

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

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

It is important to point out now that the additional states of our pedestrian - encoding vision and intention - 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 given in [8].

2.3 Definition of 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 [4].

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 similar to [4] where 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. It is important to point out that this is a mathematical model that links the measurements received from the lower level EKF to the modelled 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, it is more useful to interpret the main body axis as defining orientation, since it is usually close to that of the foot. We will introduce an angular deviation between this body orientation in space and that of the foot (IMU). This interpretation is important when we draw on additional body mounted orientation sensors such as a magnetometer. In the complete system there are 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 last three of the above, but have not explicitly represented the angles linking the last two coordinate systems (they 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. In addition, we assume that there is an additive coloured angular error between the true directional change of the person's body and that measured by the INS (which we call IMU 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" since such an animal would typically experience a large angular deviation when equipped with an IMU mounted on its foot. Since 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 IMU heading drift as random walk processes and they are formally encoded in the state variable $\mathbf{E}_k$. 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).

2.4 Map Representation in the Practical Implementation

As stated above, in our model the map is a probabilistic representation of possible human motion that is 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 behaviour in which the person looks at a probability distribution posted at each location, and "draws" the next step using exactly this governing distribution.

As mentioned earlier we resort to a RBPF that follows a FastSLAM partitioning [6] where 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 probabilistic function of the current location.

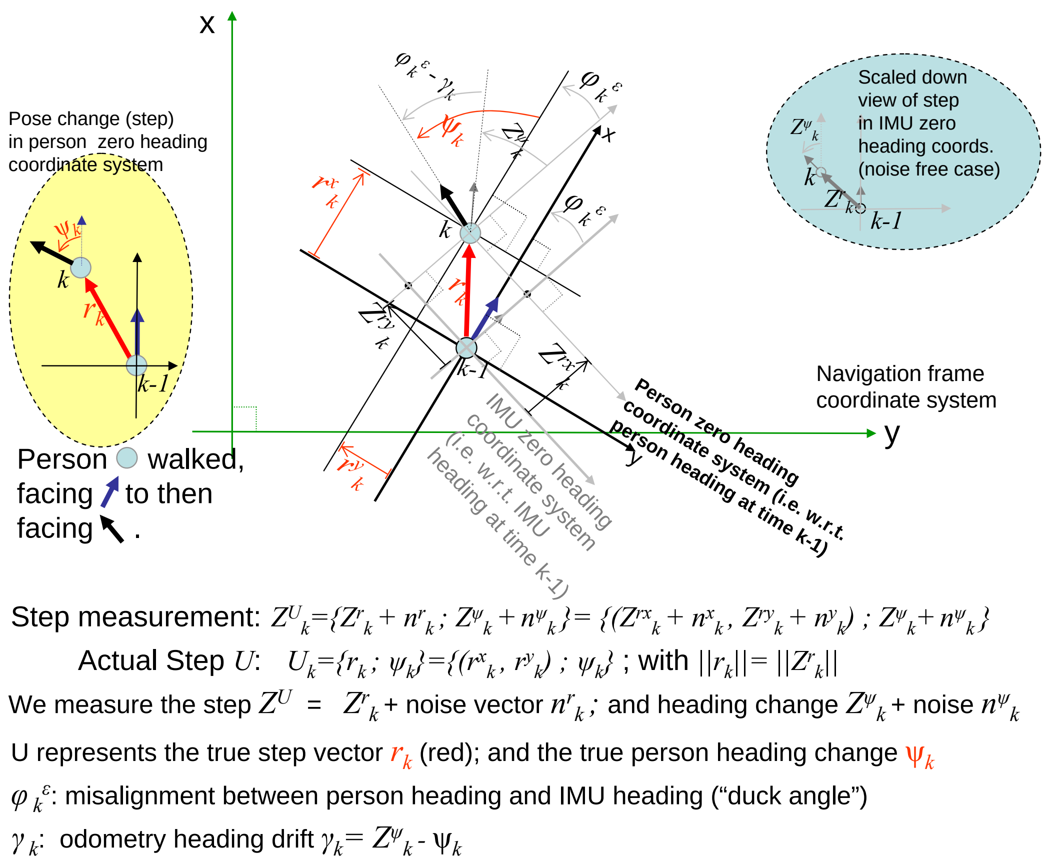

Figure 6. Representation of the important coordinate systems and step vectors (pose change) and angular errors used in our system model.

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. Furthermore, a particle explores possible deviations of the true pedestrian's path as a result of the sequence of the IMU errors $\mathbf{E}_{0:k}$ (refer to Figure 8). 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 currently occupying. As a result, particles will be rewarded by revisiting similar transitions in space.

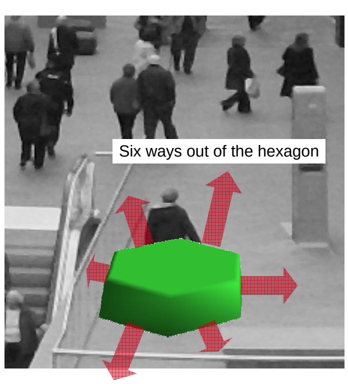

Figure 7. 2D Hexagon representation for stochastic pedestrian movement used in this paper. We represent the six probabilities for crossing each of the hexagons in a regular grid with adjacent hexagons.

We will 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 $\mathcal{H} = \{H_0, H_1, \cdots, H_h, \cdots, H_{N_H-1}\}$ as the set of N_H hexagons, where the index h uniquely references a hexagon's position. Furthermore we define $\mathcal{M}_h = \{M_h^0, M_h^1, \cdots, M_h^5\}$ as the set of transition probabilities across the edges of the h-th hexagon

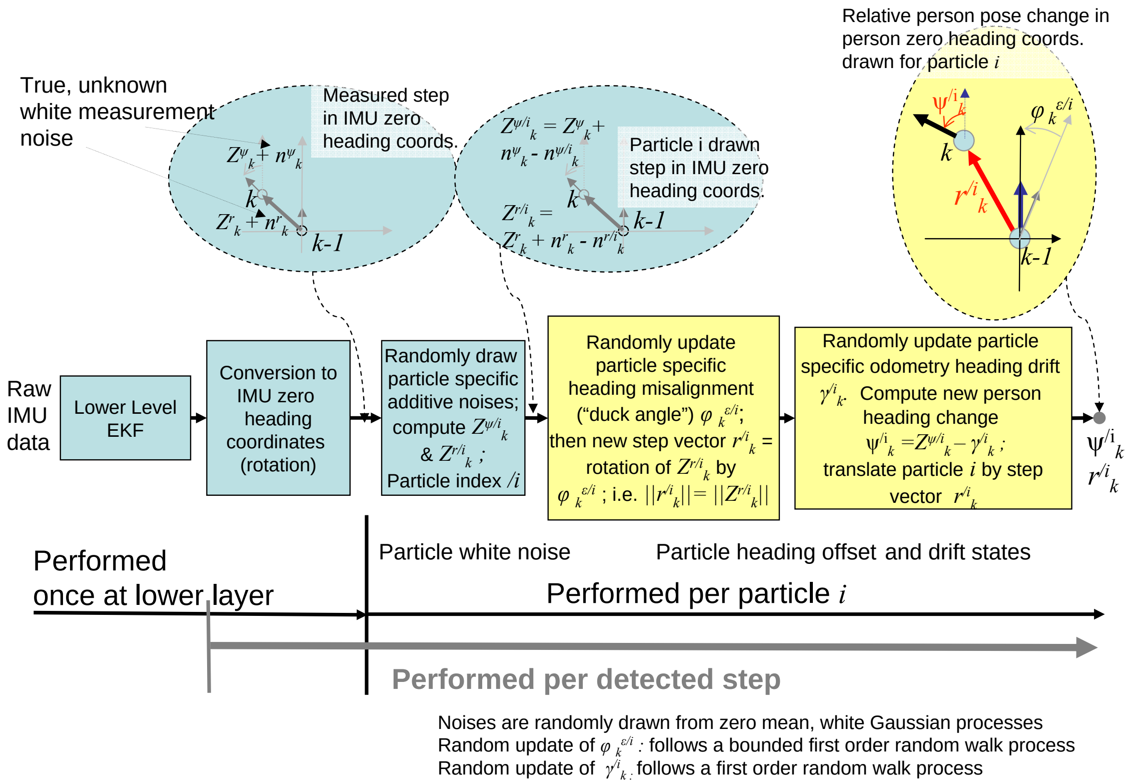

Figure 8. Processing chain from step estimates at the lower filtering level up to the stage where particles are drawn from the proposal density (5). Drawing the two odometry error state angles from their corresponding random walk processes corresponds to drawing $\mathbf{E}_k{}^i$ from $p(\mathbf{E}_k|\mathbf{E}_{k-1}{}^i)$.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

and

$$
\begin{aligned}
M_{h(\mathbf{P}_{k-1})}^{e(\mathbf{U}_k)} = \; & P(\mathbf{P}_k \in H_j | \mathbf{P}_{k-1} \in H_h) \\
& \text{s.t. } H_j \text{ is reached from } H_h \text{ by } \mathbf{U}_k, \quad (3)
\end{aligned}
$$

and $j \neq h$; i.e. we moved to a new hexagon, and where $0 <$

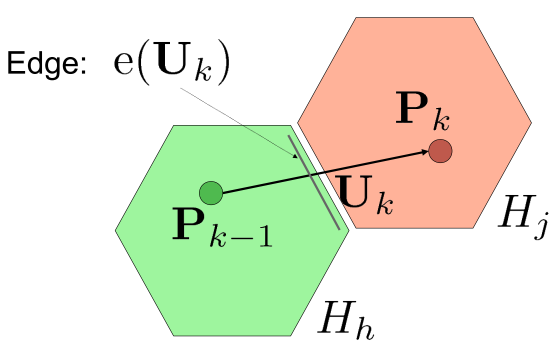

Figure 9. Definition of the hexagon transition $M_{h(\mathbf{P}_{k-1})}^{e(\mathbf{U}_k)}$ in (3).

$\mathrm{e}(\mathbf{U}_k) \leq 5$ is the edge of the outgoing hexagon associated with $\mathbf{U}_k$, i.e. the edge of the hexagon in which $\mathbf{P}_{k-1}$ lies and which borders the hexagon in which $\mathbf{P}_k$ lies - see Figure 9.

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

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

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 $\tilde{h}$ for outgoing hexagon $h(\mathbf{P}_{k-1})$, and $\tilde{e}$ for the crossed edge $e(\mathbf{U}_k)$ for brevity.

2.5 Learning the Transition Map

Learning the map on a particle-by-particle basis is very easy and is based on Bayesian learning 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 $\tilde{e}$ we count this transition in the local map of hexagon $H_{\tilde{h}}$ for particle i.

3 Summary of the Algorithm

As has been described in [4] and [2] it is advantageous to resort to a "Likelihood Particle Filter" [9] since the measurement $\mathbf{Z}_k^U$ is 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 instead of from the state transition model. Specifically, we have chosen the proposal density of the particle filter to be

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

Our RBPF algorithm operates as follows; practical issues necessary for a real implementation are explained thereafter:

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

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

   where the counts are those that are computed up to step $k-1$: The term $N_{\tilde{h}}^{\tilde{e}}$ 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 $\alpha_{\tilde{h}}^{\tilde{e}}$ and $\alpha_{\tilde{h}} = \sum_{e=0}^{5} \alpha_{\tilde{h}}^{e}$ are the priors of this map segment (in our experiments we chose $\alpha_{\tilde{h}}^{\tilde{e}} = 0.8$ for all edges and hexagons).

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

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

   (f) Resampling can be performed if required.

There are a number of implementation issues that need to be addressed in order for the algorithm to work in practice. The first of these is that 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 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. 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) to account for the fact that straight tracks with different angles traversing the grid will yield slightly different total number of hexagon edge transitions per distance travelled (otherwise particles with some directions are slightly favoured).

4 Results and Conclusions

4.1 Overview

Our results based on real data are very promising. We have collected the data of a pedestrian walking in our office environment as well as in the adjacent area outside in a number of runs lasting up to about 12 minutes each. In this paper we 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. 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 (i.e. outdoors again). In [8] we have presented results for the indoor-only case, where only a foot mounted IMU was used as sensor. In a subset of our evaluations we assumed that we knew

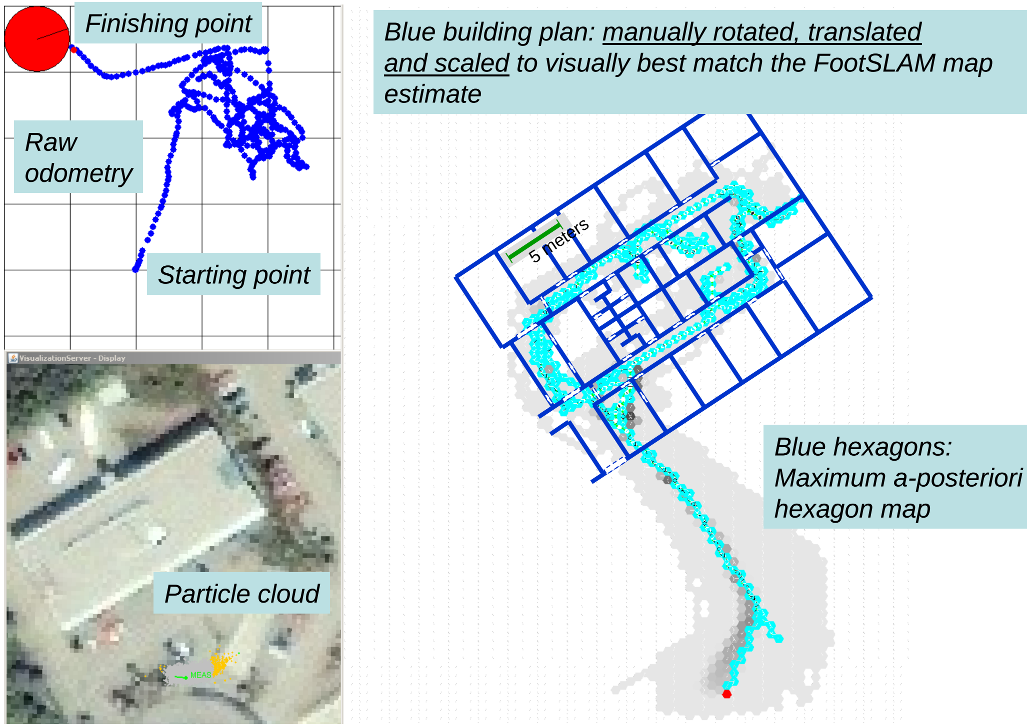

Figure 10. Result of FootSLAM processing for an outdoor-indoor-outdoor walk. Top left: raw odometry - we see how the starting and finishing point are far apart (in reality, they were very close together). Right: resulting FootSLAM map at the end of the processing run through the entire data set. Bottom left: particle cloud (note: smaller scale compared to the map window shown on right). In this experiment we assumed that we knew the building outer walls.

a-priori the location of the outside building walls to within 3 meters of the true wall locations. This helps the FootSLAM to converge a little but as we shall show, it is not a requirement. It is realistic, however, for somebody mapping a new region to roughly manually mark the outer corners of the target building 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.

In [8] the pedestrian remained indoors and followed no particular course of motion and visited a number of offices and rooms as well as the corridor area. In the work presented here, 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, a number of rooms were entered and then left, and the pedestrian continued to walk the corridor to the next room. This pattern was repeated two more times, and the building was left by the same door. In [8] we briefly discussed the sensitivity of the algorithm towards the length of time before "loop closure" (i.e. when the pedestrian retraces previous steps or areas), and the successful indoor experiments reported in [8] were partly designed to prolong the time before loop-closure.

4.2 Experiments and processing

The recorded sensor data is collected during the walk and is then processed offline in our RBPF implementation. In our visualisations, 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 (e.g. Figure 10). This allows interpretation and explanation of important effects, such as the the evolving map hypotheses; a collection of our data processing runs were recorded as video and are available under [11]. In Figure 10 and Figure 11 we show qualitative results for the cases where we assumed and did not assume rough knowledge of the building outline. In Figure 12 we show the positioning error during the walk - in comparison to other approaches.

We have also processed the three-dimensional data from [10] [11] (data set "March09Measurement03") assuming that this had been a two-dimensional measurement; this causes no major degradation as the floor plan corridors are more or less identical/compatible at different levels of the building. Obviously we ignored the altimeter data but included the magnetometer. The results are shown in Figure 13.

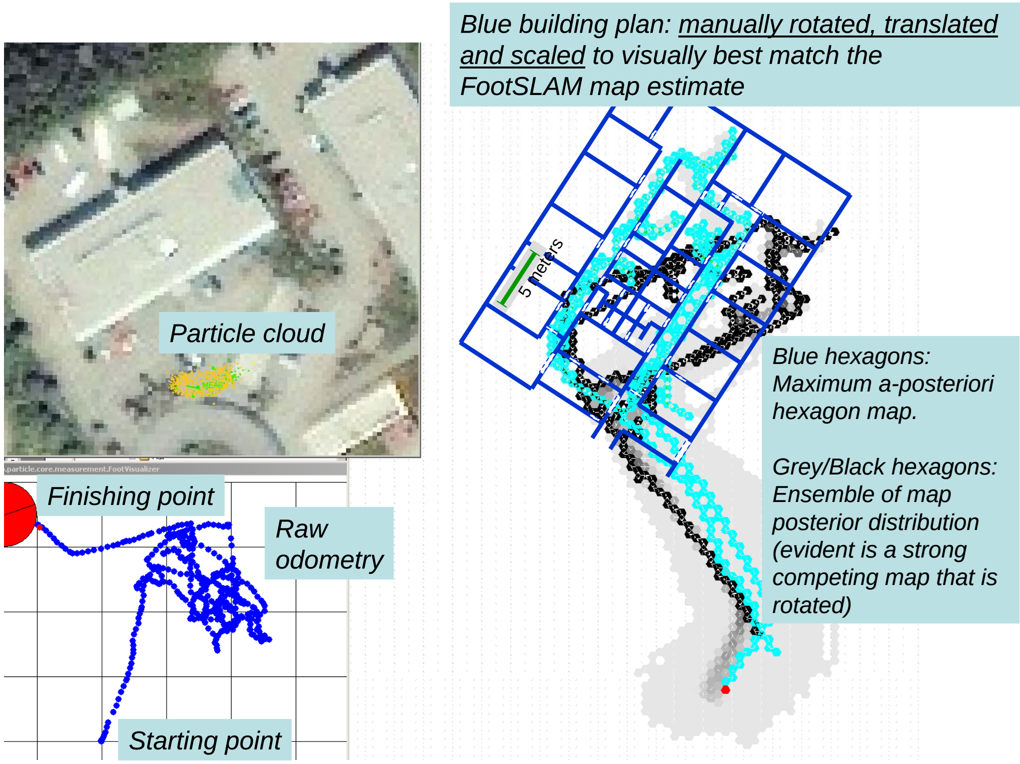

Figure 11. Result of FootSLAM processing for an outdoor-indoor-outdoor walk. Bottom left: raw odometry. Right: resulting FootSLAM map. Top left: particle cloud (note: smaller scale compared to the map window shown on right). In this experiment we did *not* assume that we knew the building outer walls.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

4.3 Discussion and further work

Comparison with the true layout of the building - which was, of course, not used in the processing and only manually inserted over the resulting FootSLAM maps applying rotation, scaling and translation chosen to match the FootSLAM map - showed a remarkable ability of the RBPF to estimate the reachable areas and errors of the location of the doors was usually to within +/- 1 meter and never more than about 2-3 meters away (based on results in this paper and on those in [8]). All results so far were obtained with just a single track 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 which 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 map combination techniques that combine maps from different sources under these considerations.

Inspecting the numerical results we 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 was 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 3-7 meters, so this is a baseline error onto which the FootSLAM relative errors are essentially added.

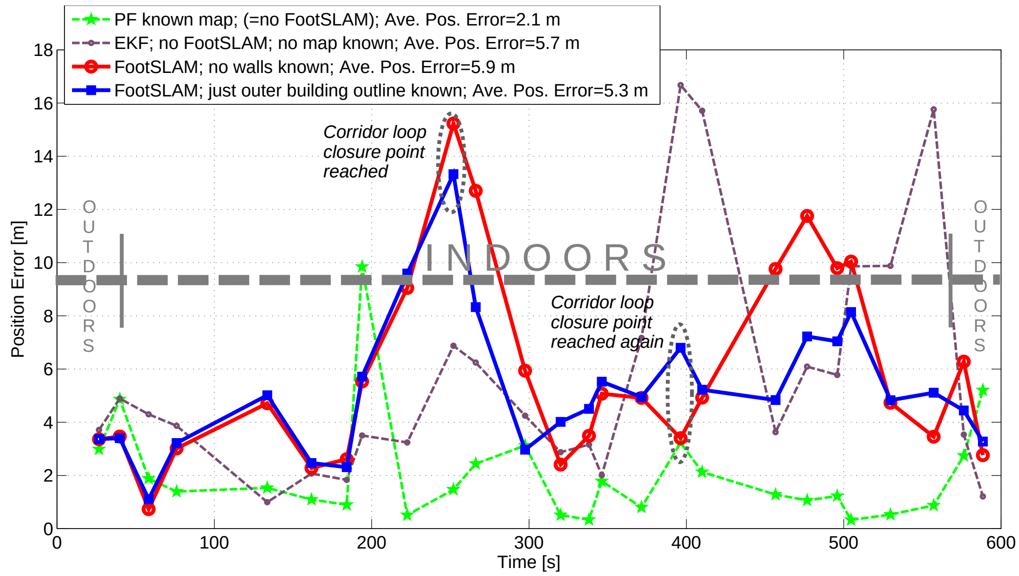

Figure 12. Position error of FootSLAM processing for the outdoor-indoor-outdoor walk of Figures 10 and 11. Comparison with particle filter using complete building plan information and a simple EKF using no such information. The FootSLAM results were averaged over three RBPF runs over the data set with 55000 particles each. The EKF and PF curves are for a single run.

- The extended Kalman filter (EKF) 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).

Since our maps are probabilistic, estimation of pedestrians' future paths could also be performed - similar to work for driver intent estimation [12]. Further work should also integrate more sensors, address 3D, as well as collective mapping where users collect data during their daily lives and maps are combined and improved.

4.4 Acknowledgements

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

References

[1] E. Foxlin, "Pedestrian tracking with shoe-mounted inertial sensors," *IEEE Computer Graphics and Applications*, vol. 25, no. 6, pp. 38–46, Nov. 2005.

[2] O. Woodman and R. Harle, "Pedestrian localisation for indoor environments," in *Proc. of the UbiComp 2008*, Seoul, South Korea, Sep. 2008.

[3] S. Beauregard, Widyawan, and M. Klepal, "Indoor PDR performance enhancement using minimal map information and particle filters," in *Proc. of the IEEE/ION PLANS 2008*, Monterey, USA, May 2008.

[4] B. Krach and P. Robertson, "Cascaded estimation architecture for integration of foot-mounted inertial sensors," in *Proc. of the IEEE/ION PLANS 2008*, Monterey, USA, May 2008.

[5] R. Smith, 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, Inc., 1990, pp. 167–193.

[6] M. Montemerlo, 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.

[7] M. Khider, S. Kaiser, P. Robertson, and M. Angermann, "A novel movement model for pedestrians suitable for personal navigation," in *ION NTM 2008, San Diego, CA, USA*, 2008.

[8] P. Robertson, M. Angermann, and B. Krach, "Simultaneous localization and mapping for pedestrians us-

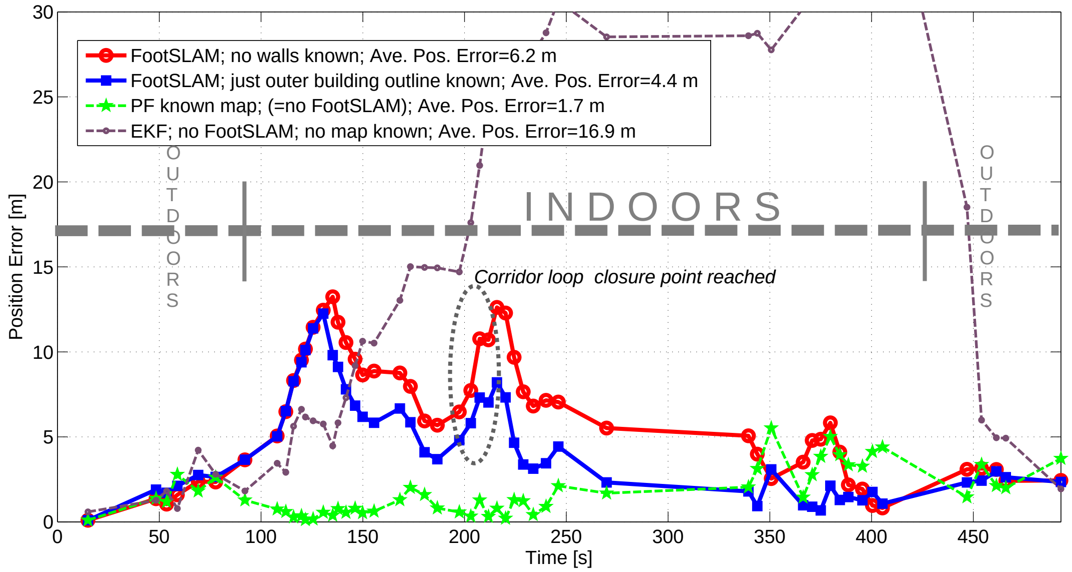

Figure 13. Position error of FootSLAM processing for the outdoor-indoor-outdoor walk of [10] [11]. Comparison with particle filter using complete building plan information and a simple EKF using no such information. Note: FootSLAM assumes this as being two-dimensional which in this originally 3D data 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.

ing only foot-mounted inertial sensors," in Proc. UbiComp 2009, Orlando, Florida, USA.

[9] S. Arulampalam, 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, Feb. 2002.

[10] M. Angermann, 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 *European Navigation Conference ENC-GNSS 2009, Naples, Italy*, 2009.

[11] "FootSLAM videos and reference data sets download," http://www.kn-s.dlr.de/indoornav.

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