Inertial-Based Joint Mapping and Positioning for Pedestrian Navigation

Michael Angermann, Patrick Robertson
Institute for Communications Technology
German Aerospace Center (DLR)
PO Box 1116
D-82234 Wessling
GERMANY

michael.angermann@dlr.de / patrick.robertson@dlr.de

ABSTRACT

In this talk we will discuss reliable and high precision pedestrian positioning in GNSS denied environments. We will begin with a brief introduction into non-linear sequential Bayesian estimation principles and their application to sensor fusion. We shall present the p9roperties of pedestrian odometry (step measurement) based on foot mounted IMUs (Inertial Measurement Units) with zero velocity updates and the nature of the resulting error process. Next, we shall introduce an approach using sequential Monte-Carlo algorithms ("particle filters") to stabilize the location estimate by the use of map information (e.g. a-priori known building layouts). Because such maps are not always available, we shall conclude with a presentation of purely inertial-based Simultaneous Localization and Mapping (SLAM) technologies and how to use them to learn the map based on pedestrian odometry data. Applications can range from offline map learning to real-time collaborative SLAM in unmapped environments.

1.0 INTRODUCTION

Accurate long-term pedestrian localization is a particularly challenging task in GPS denied environments. While deep indoor reception is often possible with high sensitivity receivers, achieving meter-level accuracy is a different matter entirely. An approach that can be followed to address this problem is multi-sensor fusion, the principle being that sensors with different characteristics can augment each other. A particularly useful sensor in this context is the inertial measurement unit consisting of one or more accelerometers and/or gyros. The advantages of inertial measurements lie above all in its resilience and autonomy and its ability to measure short-term motion (changes). Recent research in the navigation community has focussed on pedestrian dead reckoning which uses inertial sensors to measure individual human steps. The result is a relative positioning system with reasonable short-term accuracy in two or three dimensions. To achieve long term accuracy, however, without any additional aiding such as GPS or wireless positioning one can resort to (known) building layouts that allow us to constrain the pedestrian's estimated trajectory. The more a building constrains motion, the more accurate the resulting location accuracy. The main disadvantage is that an accurate building plan is required a-prior.

This contribution is structured as follows. We begin with an exposition of sequential Bayesian estimation which is the theoretical and conceptual cornerstone of the localization and mapping techniques reviewed thereafter. The main flavours of pedestrian dead reckoning are then explained and this serves as the basis for map-aided approaches. The main contribution of the paper is to motivate and introduce the concept of odometry based simultaneous localization and mapping (SLAM) for pedestrians. The objective of SLAM – a principle developed by the robotics industry- is to perform a joint estimate of maps and location based on sensor measurements, which means that maps (or building plans, in our case) are not needed a-priori. In robotics, sensors used in SLAM comprise devices such as laser scanners or cameras in addition to odometry (motion change) information from the robot. The main variant of SLAM for humans

("FootSLAM") discussed here, however, needs no sensors apart for the inertial-based human step measurement. Before sketching the mathematical foundations we shall present some important application scenarios of FootSLAM. This is followed by an example of the attainable performance accuracy. Finally, we present an augmentation of FootSLAM – "PlaceSLAM" which requires active participation of the pedestrian to improve mapping and location accuracy.

2.0 NON-LINEAR SEQUENTIAL BAYESIAN ESTIMATION

2.1 Principles

If we are interested in finding out as much as possible about the state of a system we need a measurement device that provides an output ("measurement") that is somehow influenced by the state of interest. In our application domain the state of interest might be the position of a pedestrian and the measurement device a GPS receiver, whose output hopefully depends on the actual position of the bearer. In real-world physical systems the state cannot change arbitrarily fast, due to some form of inertia. Hence, knowledge about a system's previous state tells us something about the system's current state. In order to obtain the best possible estimate of some state variable x at a time instant *k*, we should not only consider the available measurements, but also the past evolution of this state variable. If all information that we consider relevant for the future evolution of the state is represented in that state variable, say the current position and velocity for our location determination problem, we can model the evolution as a first order Hidden Markov Model as depicted in Figure 1.

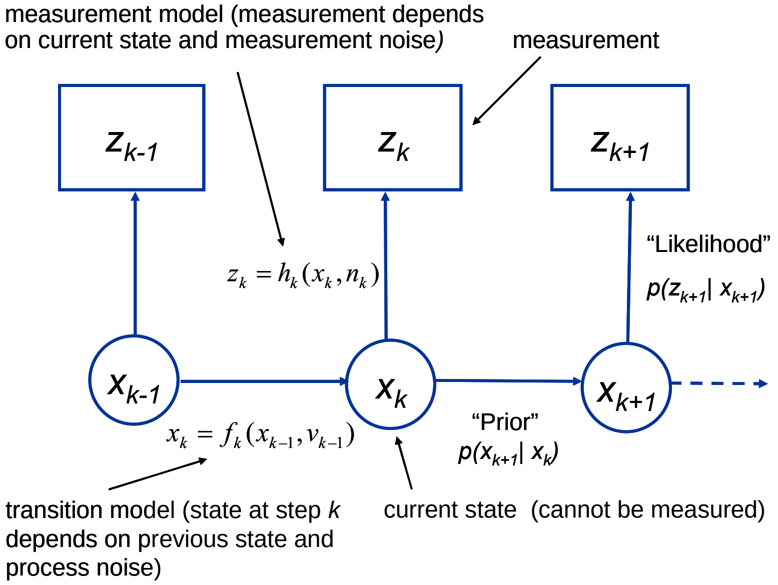


Figure 1: First order Hidden Markov Model. The current states depends only on the previous state and the current measurement depends only on the current state. In the general case, both the measurement model and the transition model may be nonlinear functions.

Predict:

replaces $p(x_k \mid x_{k-1})$

$$p(x_k \mid z_{1:k-1}) = \int p(x_k \mid x_{k-1}, z_{1:k-1})\, p(x_{k-1} \mid z_{1:k-1})\, dx_{k-1}$$

replaces $p(z_k \mid x_k)$

Update:

$$p(x_k \mid z_{1:k}) = \frac{p(z_k \mid x_k, z_{1:k-1})}{p(z_k \mid z_{1:k-1})}\, p(x_k \mid z_{1:k-1})$$

$$\int p(z_k \mid x_k, z_{1:k-1})\, p(x_k \mid z_{1:k-1})\, dx_k = \text{constant: } \alpha_\text{k}$$

**Figure 2: Derivation of the general Recursive Bayesian Filter.
The Markov condition is applied to the prediction and update step.**

Bayesian Filter – Final Posterior Equations

Transition prior

Old Posterior

$$p(x_k \mid z_{1:k-1}) = \int \underline{p(x_k \mid x_{k-1})}\, p(x_{k-1} \mid z_{1:k-1})\, dx_{k-1}$$

$$p(x_k \mid z_{1:k}) = \alpha_k \underline{p(z_k \mid x_k)}\, p(x_k \mid z_{1:k-1})$$

Likelihood

Figure 3: Final posterior equations of the Recursive Bayesian Filter.

2.2 Kalman Filter

Under the assumption that the measurement and transition functions are linear and the measurement and process noise are Gaussian, it can be shown that the Kalman filter is the optimal recursive Bayesian filter that minimizes the mean square error. Furthermore, the Kalman filter is computationally fairly efficient. The effort for each iteration is primarily determined by the cost for a matrix inversion. Today's fastest matrix inversion algorithms are $O(d^{2.4})$, for a d x d matrix. In the Kalman filter d is determined by the size of the measurement vector z_k. Interestingly and initially counter-intuitively, the covariance matrix is not a function of the measurement data and can therefore be pre-computed without taking the actual measurement data into account. It asymptotically reaches a steady state.

2.3 Extensions to the Kalman filter for Nonlinear Problems

A number of extensions and modifications to the Kalman filter exist that relax the Kalman filters assumptions with respect to the linearity and Gaussianity of its measurements and transitions. Among these modifications the Extended Kalman filter (EKF) is the "workhorse" in many applications of estimation. While the EKF is no longer an optimal filter it can be efficiently applied to a wide range of problems. Closely related to the EKF is the Unscented Kalman filter (UKF) and the Extended Information Filter. For a detailed introduction and discussion of these filters see [3].

2.4 Particle Filter

The key idea of a particle filter is to approximate the posterior density by a set of random samples ("particles"). Each particle is then associated with a scalar value ("weight"). For large numbers of samples the Random (Monte Carlo) Sampling approximation converges to the true PDF. However, the particle filter is a non-optimal approximation for any finite number of particles. The key advantage of a particle filter is its ability to incorporate nonlinear functions in its transition and measurement models as well as non-Gaussian process and measurement noise processes.

3.0 PEDESTRIAN DEAD RECKONING

3.1 Approaches

The basis for inertial based mapping for pedestrians is based on a concept from robotics: odometry, meaning measurement of a trajectory. In robotics one can view the input signals (control input) to the driving wheel motors as odometry since knowledge of these signals will allow one to estimate the change in the robot's trajectory during the application of the control input. Alternatively, one can measure the wheels' rotation rates. Due to wheel slippage one is unable to determine the actual trajectory with absolute certainty and a performing odometry with a sequence of control inputs will lead to a random walk in terms of the odometry error. In humans various approaches have been investigated to perform odometry and it is illuminating to compare these to the robotic approaches. Traditionally, most methods have attempted to detect individual steps of a pedestrian and to estimate parameters such as distance travelled and change in heading. Recently, work has been directed to actually measure the electrical "control input" to groups of leg muscles to estimate the resulting step the person took [1]. A very promising approach is to use an inertial sensor (IMU – Inertial Measurement Unit) located on the shoe. This allows true 6-DOF (degrees of freedom) inertial navigation within a relative coordinate system. The cubic error growth can be reduced to a linear error growth over time by performing a zero velocity update during the rest phase of every step [2]. 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. Researchers have also looked at other sensor placements such as inertial sensors on the head [4], knee and waist [5] or simply in trouser pockets. The difference between foot mounted approaches and those that use placements on other parts of the body is that the former perform a 6-DOF computation of the shoe's pose (position and attitude); this is in contrast to step estimation techniques which compute features in the IMU signals to detect steps and estimate their length and the direction change of the pedestrian.

We shall briefly give an overview of the approach chosen to validate "FootSLAM" – inertial based mapping for pedestrians – the foot mounted IMU and computation of the shoe's pose. We have, however, made no limitations that exclude other forms of human odometry from being used. As long as we have estimates of the length and heading change of each human step, the techniques presented here can be applied.

3.1.1 Three Dimensional Step Measurement

We shall briefly summarize the processing architecture used to drive FootSLAM with appropriate step measurements. In Figure 4 we have shown the two-layer approach presented in [11]. The gyroscope and accelerometer signals are processed in a standard inertial navigation computer at the frequency of the sensor measurements (around 100 Hz or higher). An Extended Kalman Filter (EKF) operates on the error domain of the inertial navigation computer and we have used a very simple EKF with nine error states: 3 each in position, velocity and attitude dimensions. The zero velocity update is applied when a step has been detected (see below) and is used as a velocity-error pseudo-measurement [2]. When a zero update is performed this marks the transition between one pedestrian step and the next and the pose change is output to a higher processing layer which will be described in following sections.

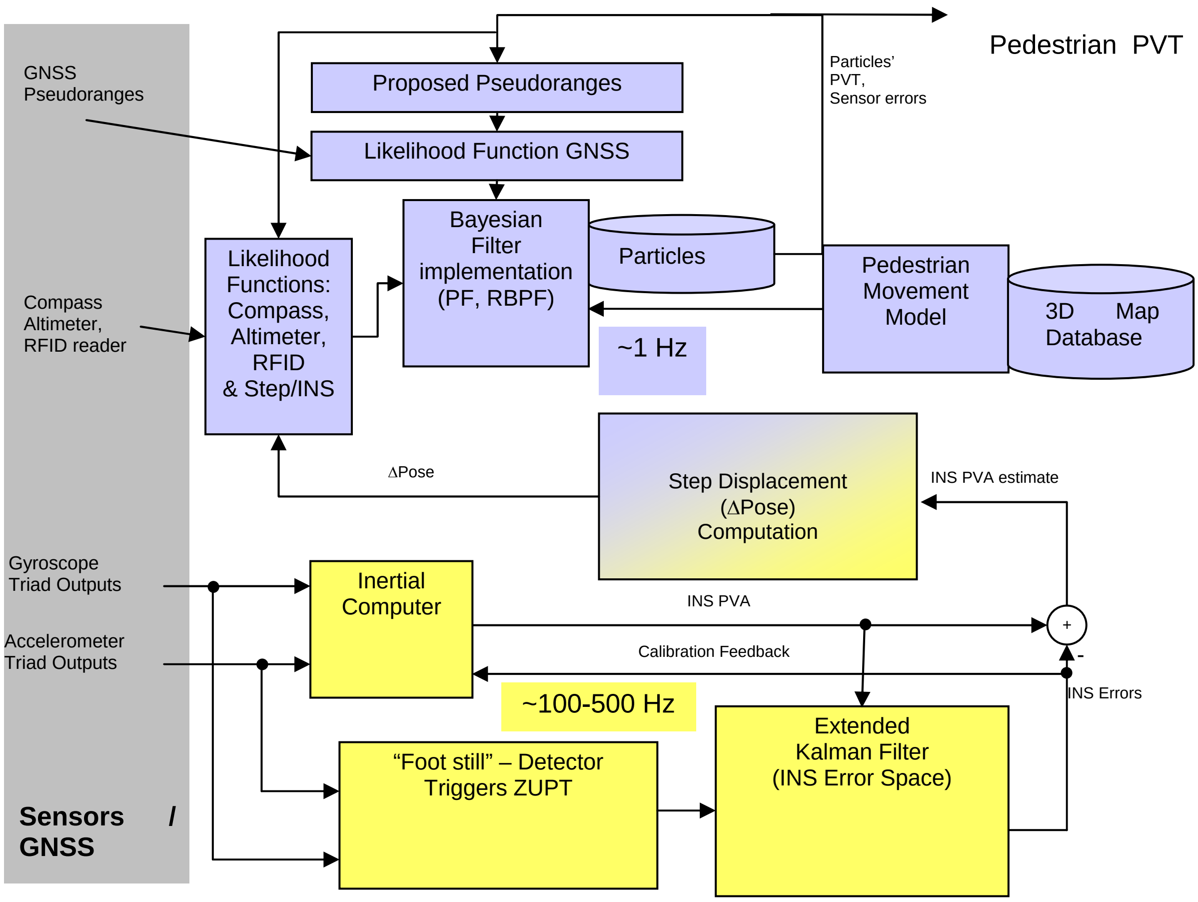


Figure 4: Processing architecture for inertial based step computation and sensor fusion.

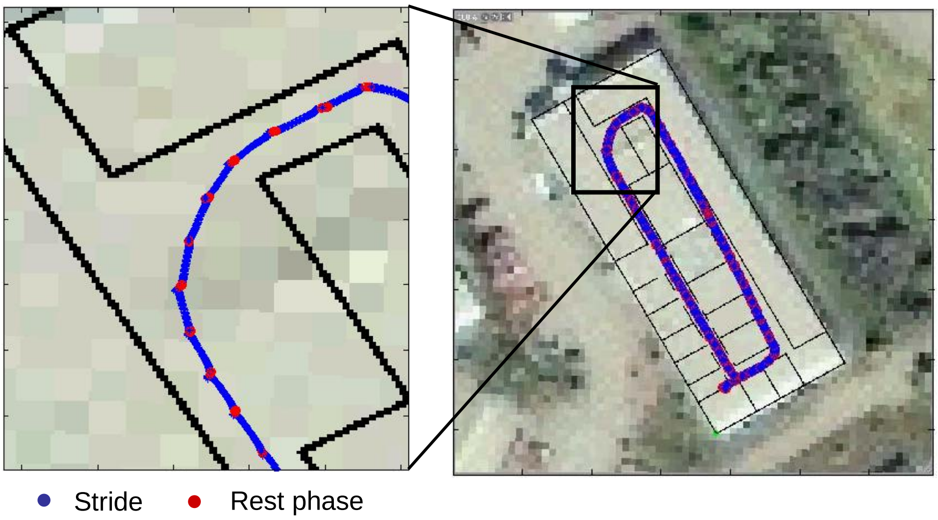


Figure 5: Trajectory of a walk around a building – EKF output with no further processing. Resulting sequence of steps was aligned with the starting pose for illustration and comparison purposes.

3.1.3 Zero Velocity Updates

As mentioned above we can use foot-mounted inertial measurement units themselves to provide zero velocity updates — ZUPTs — during the rest phase of a pedestrian's stride.

The zero update tells us when the step has been completed (resting phase of the foot, see Figure 6) and enables us to estimate some aspects 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. In [6] we have described and compared different simple ZUPT algorithms.

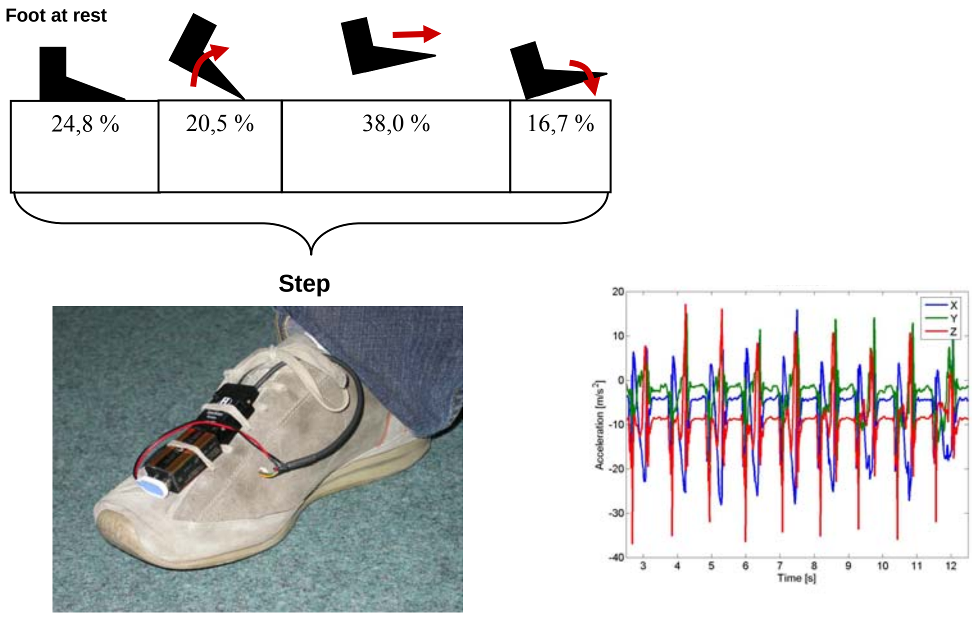


Figure 6: Sequence of human steps and an exemplary plot of the accelerometer outputs from a shoe-mounted IMU.

3.2 Error Characteristics

We have carried out a number of experiments to analyze the error characteristics of Pedestrian Dead Reckoning. For this purpose a ground truth trajectory has been determined with an optical motion capture system. The optical tracking system is not subject to relevant drift errors and provides between one and two orders of magnitude better positioning accuracy than the pedestrian dead reckoning under test.

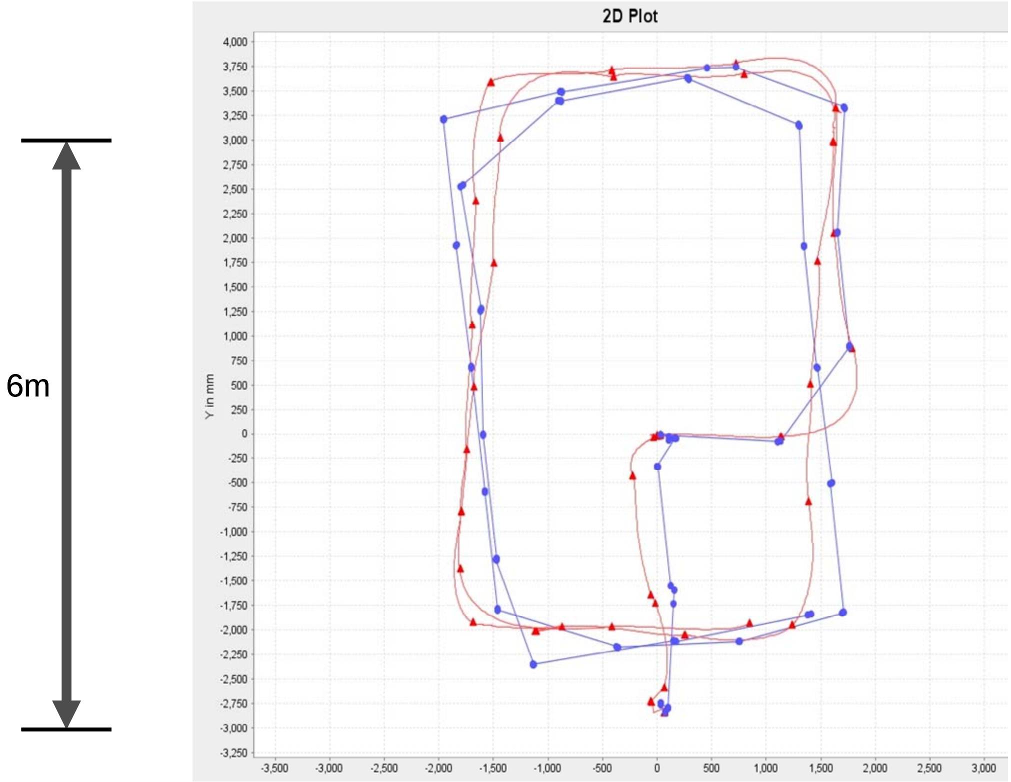


Figure 7: Estimated trajectory by a an EKF filter (blue) and ground truth obtained via an optical motion capture system (red) of a walking pedestrian with a foot-mounted IMU. Detected steps are represented by blue dots. Red triangles mark the true position for ground truth positions with corresponding timestamps.

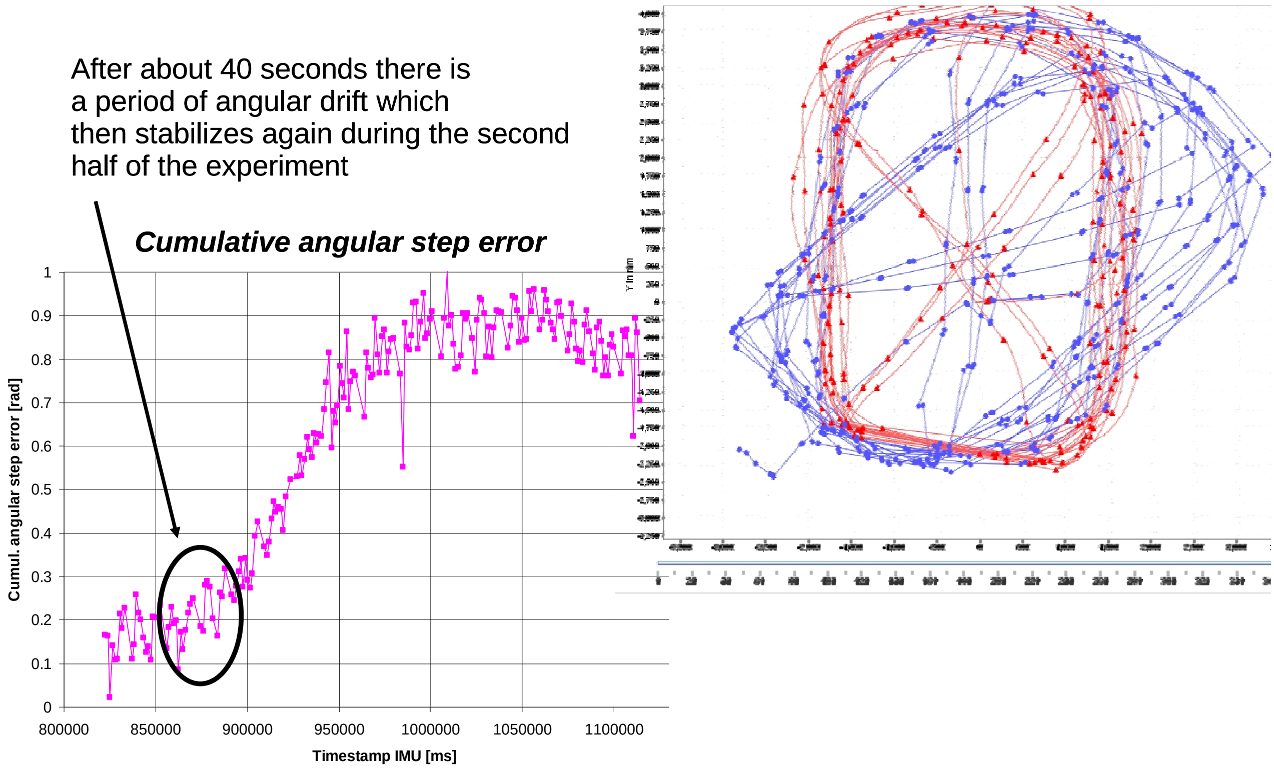


Figure 8: Estimated trajectory by a an EKF filter (blue) and ground truth obtained via an optical motion capture system (red) of a walking pedestrian with a foot-mounted IMU. Detected steps are represented by blue dots. Red triangles mark the true position for ground truth positions with corresponding timestamps. Walk duration was approximately 3 minutes. Clearly visible is the catastrophic effect of an angular drift that occurs 40 seconds into the experiment.

4.0 MAP-AIDED PEDESTRIAN DEAD RECKONING

4.1 Motivation

Dead reckoning is generally subject to unbounded error growth over travelled distance and to some extend time (when resting phases are detected, the integration can be stopped and error only is incurred during movement phases). This unbounded error growth is unavoidable in open space. However, when the environment sufficiently constrains the movements and these restrictions are known, it has been shown that the error can be limited. Figure 9 shows a comparison between a filter that ignores the building layout, and therefore incurs significant and growing error, and a filter that utilizes the knowledge about the layout, i.e. the map.

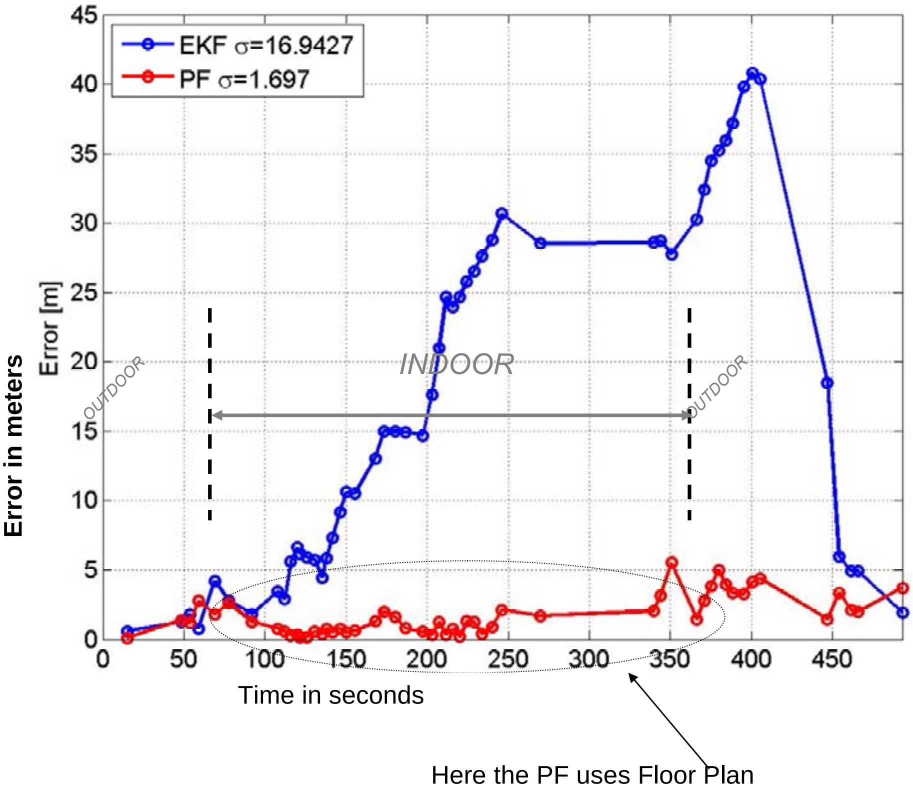


Figure 9: Influence of map knowledge on positioning error. In this experiment GPS coverage is available in the beginning and in the end (outdoor phases). During the indoor phase the position is determined solely from measurements provided by a foot-mounted IMU and building layout information. While After approx. 5 minutes in the building the error builds up to almost 30 meters if no map is used, whereas the error remains consistently below 5 meters when the map is used.

The map's value in terms of its error limiting effect, becomes even clearer in Figure 10, where a map is used to decrease uncertainty from a uniform a priori distribution over the entire building, via a strongly multimodal probability density to finally a multimodal probability density that tracks the pedestrian.

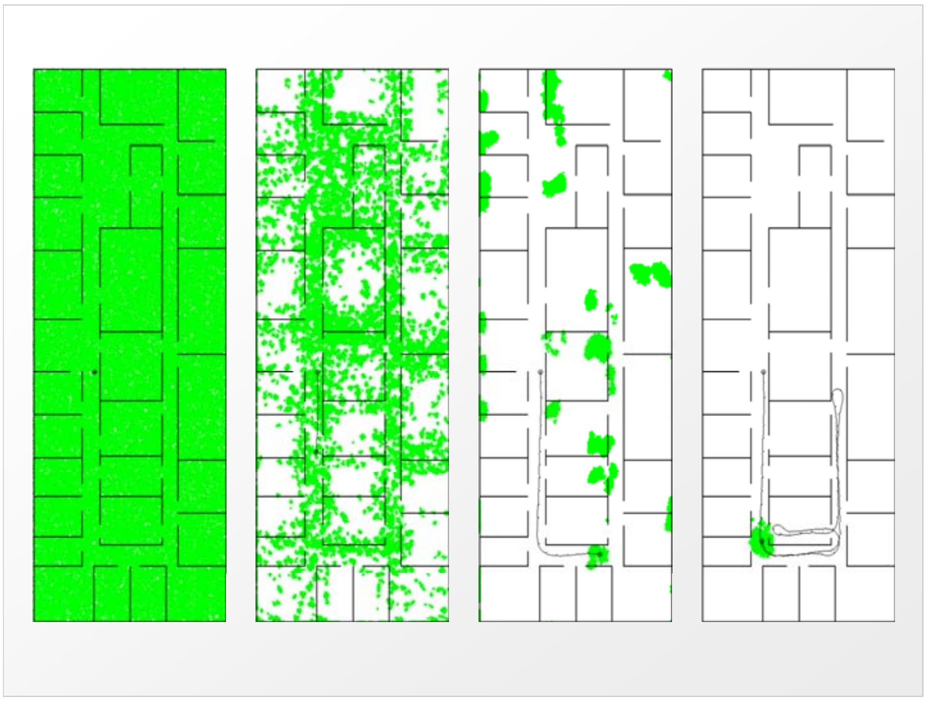


Figure 10 Integration with map-matching in the particle filter: A pedestrian wearing a foot-mounted sensor walked the indicated track (black line). At each figure the posterior position estimate (green) becomes increasingly accurate, after 80s it is unimodal.

5.0 SLAM BASED PEDESTRIAN DEAD RECKONING

5.1 Motivation for the Learning of Maps

The physical environment constrains the pedestrian's movement. We have seen in the previous section how valuable information about this environment, typically encoded in a map, can be for the determination of the pedestrian's position. If properly utilized, the map not only increases the accuracy of the location determination but also reduces computational complexity by ruling out hypotheses that conflict with the map. Without the map a particle filter would need to consider, i.e. represent these hypotheses which would consume computational resources in terms of system memory and processing. Additionally, maps per se may have value for better informing the user (via an appropriate user interface) on his current status or the environment or as the basis for routing the user through a building.

5.1.1 Usage Scenarios

A multitude of usage scenarios exist for SLAM based pedestrian dead reckoning, both in the consumer and the professional realm. We give three prototype scenarios that shall serve to illustrate how SLAM based pedestrian dead reckoning may be used under various circumstances and that are intended as building blocks for applying the scheme to a variety of other scenarios with similar characteristics. Common to all three is the assumption that no up-to-date and accurate map information of the building structures exist or are not readily accessible. Despite the existence of building plans in some form, be it in paper or electronic format, such plans are rarely reflecting the current state of the building in terms of its relevant features that facilitate or constrain movement, such as furniture, displays or dry-wall structures. Furthermore, it is unlikely that in the near future up-to-date building plans will be available for all buildings in the required electronic format within minutes or even seconds, which for some usage scenarios, such as in law enforcement or emergency response may be the relevant timeframe.

5.1.1.1 Rapid Mapping of Buildings

The most straightforward usage scenario is the rapid mapping of buildings without any prior information on their internal layout. In this scenario we wish to derive a map that will then serve as basis for localizing pedestrians by map-aided pedestrian dead reckoning. For this purpose one pedestrian needs to roam through all accessible rooms and areas on all levels of a building. The pedestrian carrying out the mapping task needs to be equipped with some form of odometry-generating sensor, such as a foot-mounted inertial measurement unit (IMU) shown in Figure 6. In this scenario the measurement data needs to be recorded and will then be processed offline. The resulting map is then stored at a server or distributed to localization devices that use it to perform map-aided pedestrian dead reckoning. Since it is not necessary to carry out specific measurements, such as setting up surveying equipment, no dwell time during mapping occurs and the duration of the entire mapping process is determined by the path length and the walking speed.

5.1.1.2 Refining and Updating Maps of Buildings

Probably the most frequent scenario will involve some existing map of a building, which may either stem from a rapid mapping as described in the previous section or information on the basis of CAD-derived building plans. In this case deviations from the map and the true layout become observable in the measurement data generated by pedestrians that are using their devices to perform map-aided pedestrian dead reckoning. Repeatingly occurring conflicts between their trajectories and the map are detectable. In such cases their measurements can be easily integrated in an offline map generation process that then combines the measurement used for the previous map and the latest measurements. However, it is important to be aware of the fact that detectable conflicts are newly passable regions that were considered to be blocking the pedestrians’ movement in the old map. Detecting newly blocked regions is more difficult and would involve some form of aging the measurements, i.e. discarding older measurements (that would hint am area to be passable). Fortunately, the easily detectable discrepancies are the critical ones that may lead to tracking loss or reduced accuracy (for a more detailed discussion on the effect of map errors on map-aided pedestrian dead reckoning see [8]).

5.1.1.3 Real-Time Collaborative Mapping of Buildings

The most challenging scenario is the mapping of a building by multiple collaborating pedestrians with the objective to provide real-time map and position information of all collaborating pedestrians. Such a scenario may occur in emergency situations were multiple teams of fire-fighters enter a building through the same of different entrances and carry out search and rescue tasks and want to avoid unnecessarily revisiting areas or involuntarily leaving out areas. Figure 11 shows a training situation were members of a SWAT team enter a building about which they may have no prior layout knowledge. In law enforcement applications, accurate determination of every team member’s position and providing this information on a map may significantly improve mutual situation awareness and potentially reduce the risk of accidentally harming a team member. In this application the real-time requirements may be severe and no a priori map data may be available. Hence, the collaborative mapping needs to be carried out and displayed in real-time with positions estimates (and the respective uncertainties) of all team members.

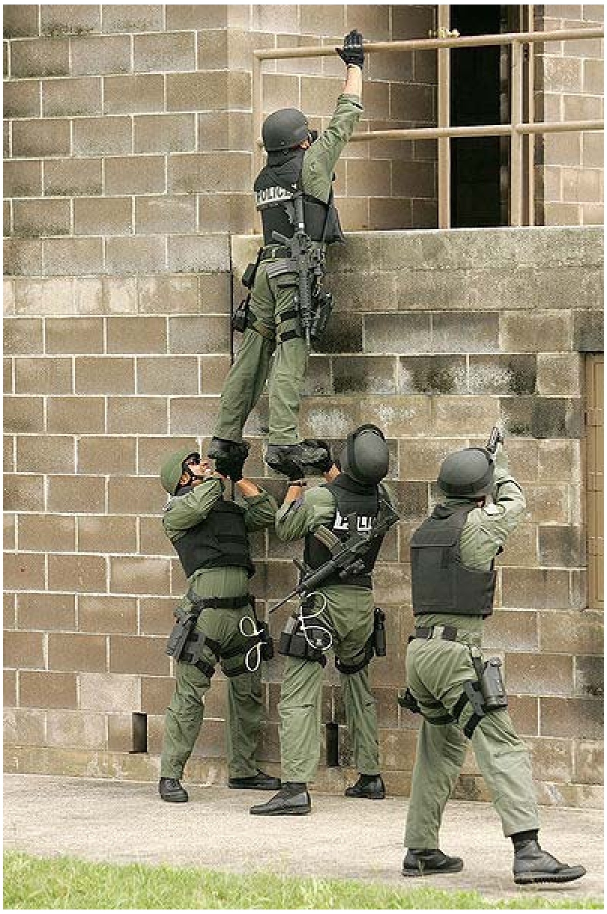


Figure 11: Members of the 37th Training Wing's Emergency Services Team entering a target building during training (U.S. Air Force photo by Robbin Cresswell).

5.2 Brief Introduction to 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. In SLAM, a moving robot explores its environment 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.

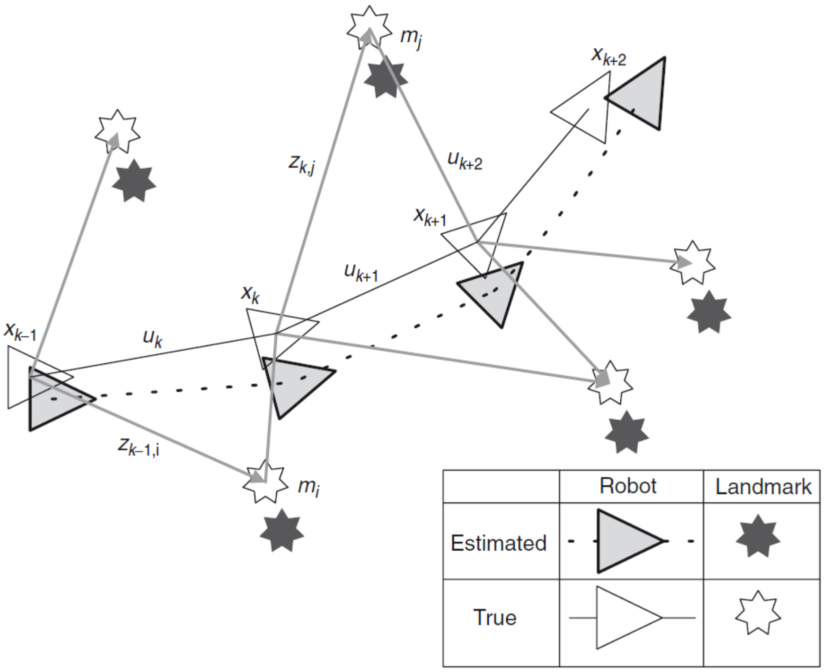


Figure 12: The essential SLAM problem. A simultaneous estimate of both robot and landmark locations is required. The true locations are never known or measured directly. Observations are made between true robot and landmark locations (illustration and caption from [19]).

There are two main strands 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. Since the state vector needs to be augmented with every additional landmark, the covariance matrix needs to grow correspondingly. An alternative 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. The conditioning on a pose allows the landmarks to be estimated independently. 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. For a very thorough introduction into SLAM see [10].

5.3 Principles of FootSLAM

We now present a new pedestrian localization technique that builds on the above explained principle of simultaneous localization and mapping. 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 – but it will work with any other kind of human odometry.

In contrast to SLAM used in robotics, FootSLAM does not require specific feature–detection sensors, such as cameras or laser scanners. Using this technique, 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 a offline fashion in order to provide maps for other people.

Human motion is a complex stochastic process, and one needs to model it in a sufficiently simple fashion in order to develop a 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. In order to understand the concept of the kind of map that FootSLAM can generate let us consider the following situation: An able-sighted person (for our purposes, equipped with foot mounted sensors) is standing inside a shopping center facing a wide area in front of him. 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 there is, of course, no direct access to the visual cues that our subject sees. However, one does have noisy measurements of the resulting motion, i.e., the steps taken from an initial position. Therefore, one can indirectly and very loosely observe these physical constraints and visual features/cues by observing the pedestrian's steps as measured by the foot-mounted IMU.

In principle one could take one of two approaches to assess possible human motion at a particular location: either interpret the local 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.

FootSLAM follows the second approach and currently limits the associated Markov process to just a single step (first order). In other words, it represents the possible future next step of the subject based only on their current location, and learns the probabilities of each possible next step through observation. This would be simple enough given 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, the FootSLAM derivation follows FastSLAM approach [9] 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. The algorithm converges remarkably quickly as long as the person revisits locations once or twice during the estimation process.

A realistic derivation needs to be 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 13).

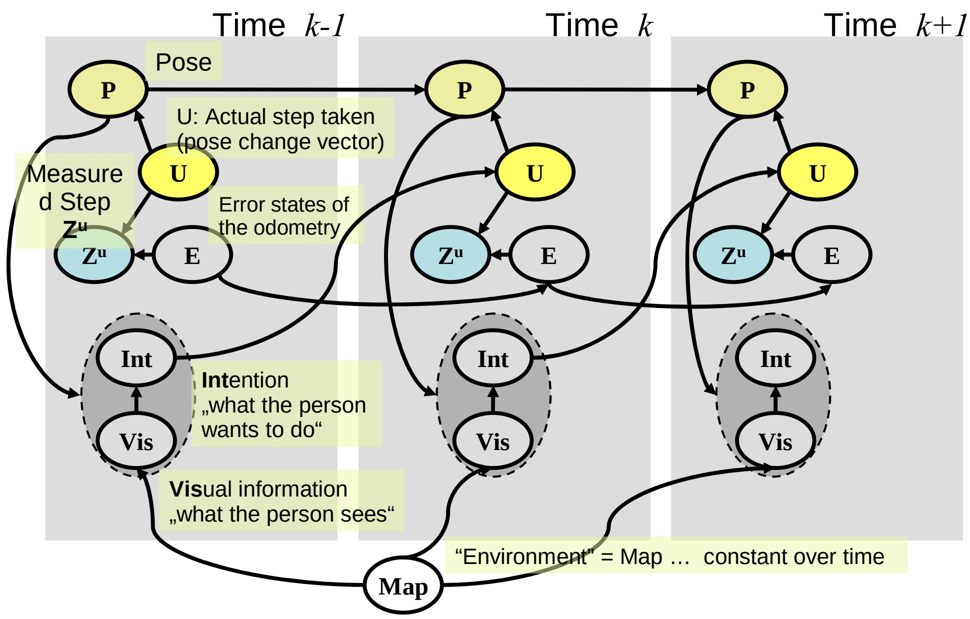


Figure 13: Dynamic Bayesian Network (DBN) for FootSLAM showing three time slices and all involved state (random) variables. The *map* can include any features and information to let the pedestrian choose their Intention *Int*. This DBN is the basis for the formal derivation of the Bayesian filtering algorithm.

In this DBN we see the following nodes (random variables, written in bold):

- *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. Note 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 one 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 third variable.
- *Inertial sensor errors* $\mathbf{E}_k$: all the correlated errors of the inertial system, for instance, angular offsets or drifts.
- *Step measurement* $\mathbf{Z}^U_k$: A measurement subject to correlated errors $\mathbf{E}_k$ as well as white noise. See Figure 14 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 $\mathbf{Int}$.

5.3.1 FootSLAM Sensors: The Step Measurement

In this section we will show details on how we represent the step transition vector between two steps that a person takes and also discussed in further detail in [21].

In order to separate the process of updating the inertial computer driven by the IMU and the ZUPTs from the overall SLAM estimation, it is advantageous to resort to a two-tier processing in which a low-level extended Kalman filter computes the length and direction change of individual steps (see Figure 4). This step estimate is then incorporated into the upper level particle filter in the form of a measurement. Note that this is a mathematical 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.

A *step* is defined 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: it is assumed that 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 the FootSLAM and dead reckoning context, however, it is more useful to interpret the main body axis as defining orientation, because this axis is usually close to that of the foot.

5.3.2 FastSLAM Factorization

The 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,

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

which following the RBPF particle filtering approach one can factorize into

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

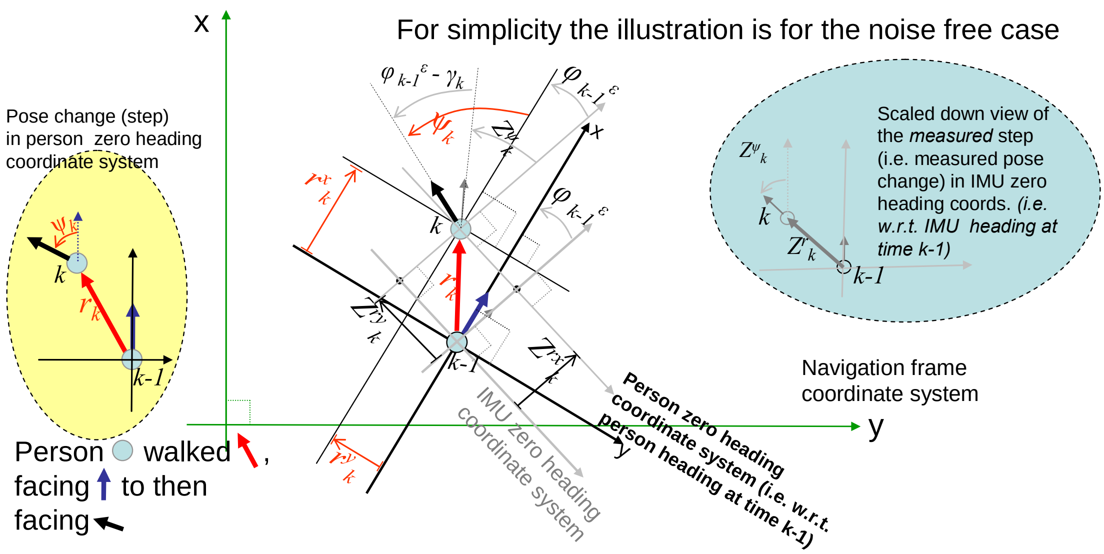
One must emphasize that the additional states introduced in the above DBN — encoding vision and intention — of the pedestrian are never actually used; they only serve as important *structural* constraints in the DBN (linking $\mathbf{P}_{k-1}$ and $\mathbf{M}$ as "parent" nodes of $\mathbf{U}_k$).

5.3.3 Step and Global Coordinate Systems

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 14 we show the final three of these coordinate systems but have not explicitly represented the angles linking the last two (which are trivial). It is assumed that the step measurement suffers from both additive white translational noise and white noise on the estimated heading change of the IMU.



Step measurement: Z^U_k = Z^r_k + noise vector n^r_k; and heading change Z^ψ_k + noise n^ψ_k

$$Z^U_k=\{Z^r_k+n^r_k\,;\,Z^\psi_k+n^\psi_k\}=\{(Z^{rx}_k+n^x_k,\,Z^{ry}_k+n^y_k)\,;\,Z^\psi_k+n^\psi_k\}$$

Actual (unknown) step U: $U_k=\{r_k\,;\,\psi_k\}=\{(r^x_k,\,r^y_k)\,;\,\psi_k\}$; with $||r_k||=||Z^r_k||$

U represents the true step vector r_k (red); and the true person heading change ψ_k

$\varphi_{k-1}^{\varepsilon}$: misalignment between person heading and IMU heading ("duck angle"); @ time *k-1*

γ_k: odometry heading drift $\gamma_k = Z^\psi_k - \psi_k$; i.e. the noise free part of the angular error

Figure 14: Coordinate systems and step definition used in FootSLAM. The misalignment angle as well as the odometry heading drift constitute the state variable E_k.

Moreover, it is assumed that an additive colored angular error appears between the true directional change of the person's body and that measured by the INS (which is called *IMU heading drift*). Finally, the model includes a very slowly changing angular offset between the person's body heading and IMU heading — for illustrative purposes it called 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 the additive noise components are modeled as being white, they do not form a part of the error state of the odometry. The "duck angle" as well as the odometry heading drift are modeled as random walk processes and are formally encoded in the state variable $\mathbf{E}_k$. Hereby the IMU heading drift is unbounded but the "duck angle" random walk process is restricted to ±20 degrees (essentially limited by human physiology).

5.3.4 Probabilistic Map Representation

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.

A Rao-Blackwellized particle filter (RBPF) that follows the above FastSLAM partitioning lets 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 human motion is represented as a first-order Markov process: the next step taken by the pedestrian is solely a probabilistic function of the current location.

In present realizations of FootSLAM the space (restricted so far to two dimensions) is partitioned into a regular grid of adjacent and uniform hexagons of a given radius (e.g., 0.5 meters).

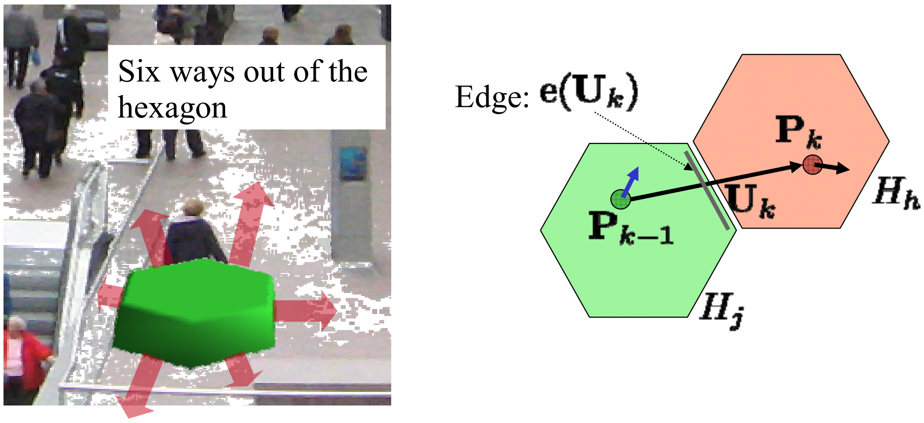


Figure 15: Definition of the step U_k and two adjacent hexagons in the FootSLAM map.

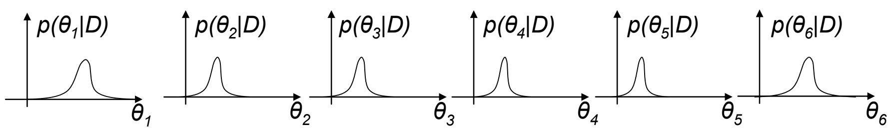
Every particle holds (conditional) estimates of the transition probabilities across the edges of all visited hexagons and updates these based on the motion taken by the particle hypothesis. A particle explores possible deviations of the true pedestrian's path as a result of the sequence of the IMU errors $\mathbf{E}_{0:k}$ (refer to Figure 14). When using a large number of particles (N_p) in the particle filter the particle cloud is exploring a very large space of possible odometry errors and at least one particle will usually be very close to the true odometry error sequence. The larger N_p, the more reliable FootSLAM becomes.

A loaded dice follows a **Discrete Distribution:**
t is the outcome of the dice roll (1...6)

$$P(\mathbf{t} = i | \Theta_i = \theta_i) = \theta_i$$

This means that the dice is completely characterized by the set of **physical probabilities** θ_i with $1 \le i \le 6$, and θ_i summing to unity.

Suppose that we have now observed some rolls of the dice (or human transitions across a hexagon edge), our observed counts *D*. The parameters of the dice, the **physical probabilities** θ_i, now *individually* follow a ***Beta*** distribution (assuming a Dirichlet prior):



The parameter *vector* θ, i.e. the vector of six physical probabilities, however, follows a ***Dirichlet*** distribution.

The more rolls of the dice / transitions we observe the more the Dirichlet distribution of θ will tend to a dirac distribution.

Figure 16: The analogy between learning the probabilities of a loaded dice and learning the transition probabilities of a hexagonal FootSLAM map.

The following assumes a two-dimensional position domain populated with hexagons of radius r. One can restrict this space to the region visited by any particle and define the hexagon H_h where the index h uniquely references a hexagon's position. Furthermore, $\mathbf{M}_h = \{\mathbf{M}^0_h, \mathbf{M}^1_h, \cdots, \mathbf{M}^5_h\}$ is the set of transition probabilities across the edges of the h-th hexagon and the map is defined as:

$$\mathbf{M}^{e(\mathbf{U}_k)}_{h(\mathbf{P}_{k-1})} \hat{=} P(\mathbf{P}_k \in H_j | \mathbf{P}_{k-1} \in H_h) \text{ ; where } j \neq h,$$

so the step $\mathbf{U}_k$ moved from H_h to a new hexagon H_j , where $0 \leq e(\mathbf{U}_k) \leq 5$ is the edge of the outgoing hexagon associated with this step, 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 right part of Figure 15). When $\mathbf{M}^e_h$ is written in boldface we are denoting a random variable. This underlines the notion that $\mathbf{M}^e_h$, a probability, is unknown. FootSLAM estimates $p(\mathbf{M}^e_h|\mathbf{P}_{0:k-1})$ as the result of observations of the sequence of positions up to step k. In the following we will use the tilde symbol ~ with h for outgoing hexagon $h(\mathbf{P}_{k-1})$, and with e for the crossed edge $e(\mathbf{U}_k)$ for brevity.

Learning the transition map on a particle-by-particle basis is very easy and is based on a Bayesian inference of multinomial and binomial distributions based on observations of a random variable (see Figure 16 for an explanation of terms). Each time a specific particle with index i makes a transition $\mathbf{P}_{k-1}^i \rightarrow \mathbf{P}_k^i$ across hexagon edge e one counts this transition in the local counter store of the pertinent hexagon for particle i and this akin to making one observation of a (loaded) dice roll in Figure 16.

5.3.5 Algorithm Summary

We can take good advantage of a *Likelihood Particle Filter* [12] because the step measurement $\mathbf{Z}^U_k$ is very accurate [11]. Weighting with a "sharp" likelihood function $p(\mathbf{Z}^U_k|\mathbf{U}_k, \mathbf{E}_k)$ would cause most particles outside the measurement to receive very low weight and effectively be wasted. Thus, it is better to sample using the likelihood function rather than sampling from the state transition model. The particle filter is summarized in Figure 17 and Figure 18.

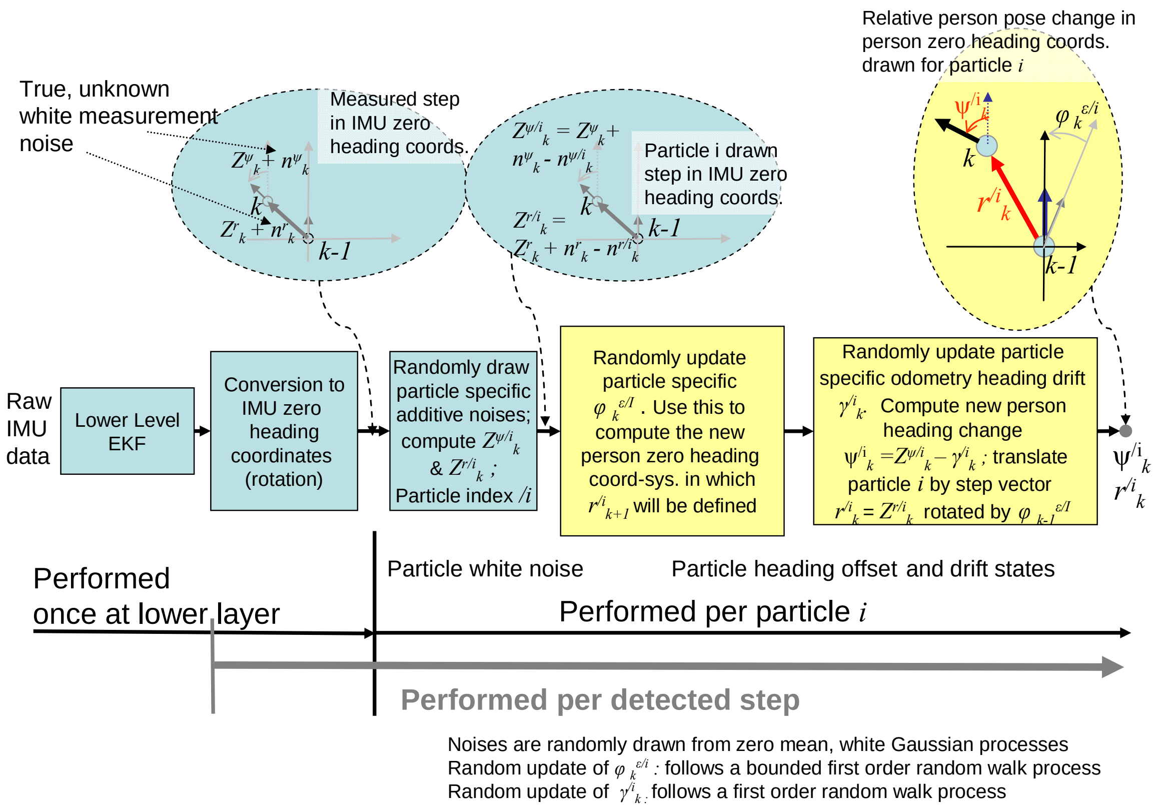


Figure 17: Processing in FootSLAM at the per-step and per-particle levels. The lower layer processing is completed once per (pedestrian) step and this initiates all particles to be processed. The notation /i refers to particle *i*.

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) Particles are drawn from the proposal density $p(\mathbf{E}_k|\mathbf{E}_{k-1}{}^i)\cdot p(\mathbf{U}_k|\mathbf{Z}_k^U, \mathbf{E}_k{}^i)$.

   (b) The pose $\mathbf{P}_k{}^i$ is computed by adding the vector $\mathbf{U}_k{}^i$ to $\mathbf{P}_{k-1}{}^i$.

   (c) The particle weight updates are simply

   $$w^i \ \alpha \ w_{k-1}^i \cdot \left\{ \frac{N_{\tilde{h}}^{\tilde{e}} + \alpha_{\tilde{h}}^{\tilde{e}}}{N_{\tilde{h}} + \alpha_{\tilde{h}}} \right\}^i$$

   where the counts are those that are computed up to step $k-1$.

   (d) 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 $\mathbf{P}_k{}^i$.
   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 (**??**) the next time $H_{\tilde{h}^i}$ is visited by this particle.

   (e) Resampling can be performed if required.

Figure 18: Summary of the FootSLAM algorithm.

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, one assumes that observing a certain transition from an outgoing hexagon to an incoming one allows the algorithm 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 one should not waste this information.

Next, it is 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 the particle in the hexagon or it might lead it over several. To address this one simply performs a weight update only when the step leads out of the last hexagon and applies multiple products in the weight update (6) for all edges crossed if the step was a larger one. Consequently, one needs to update the counts of all edges crossed between $\mathbf{P}_{k-1}{}^i$ and $\mathbf{P}_k{}^i$.

It is also necessary to incorporated a small correction term in the weight update equation in step 2.c. of the algorithm (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 travelled. (Otherwise, particles with some directions would be slightly favoured.)

5.4 Experimental Results

FootSLAM has been verified for INS-alone [20] and in combination with GPS [21]. When combined with GPS the pedestrian 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. 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 quantitative evaluation of the achieved localization accuracy. Figure 19 shows an example of a FootSLAM map learnt after about 12 minutes of walking inside the building. During the processing with the particle filter the estimate start converging once the pedestrian backtracks or revisits a region for about 10 meters. Hexagons have to be re-visited once or twice in each main undirected traversal axis for a usable map to emerge, and this governs the required duration of a walk. Accuracy will be limited by the average physical structure dimension, such as corridors and doors which is about 1-2 m.

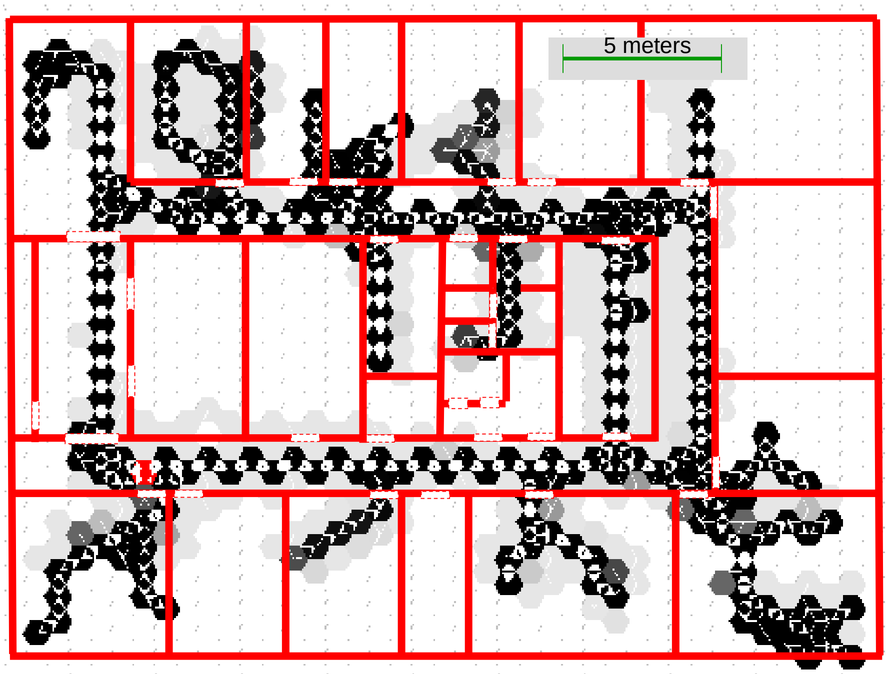


Figure 19: An indoor FootSLAM hexagon map learnt from foot mounted IMU data alone plotted on the actual building layout for reference; see [20] and for further details.

To obtain quantitative error assessments of FootSLAM we recorded ground truth location for two positions at opposite corners of the main corridor of our test building. The error growth (as the walk progressed) for FootSLAM processing is shown in Figure 20. Our coordinate system origin was both the starting point and one of the reference points and we restricted the hexagon area to remove rotation ambiguity With sufficient particles FootSLAM achieves an accuracy of relative position to within two meters at the two reference points. Without FootSLAM one sees error growth after some time - the INS “coasted” without too much error for about 300 seconds. As discussed earlier such periods of little odometry error change range typically from 30s - 300s and suggest that without maps the PF can bridge areas like large halls where there are no features for FootSLAM to map; at least to let someone find a destination or till a restricting door/corridor is again reached. In order to work, we expect FootSLAM to need a certain minimum average restrictedness of motion (related to entropy) but it can survive some open areas given sufficient particles. To achieve accurate mapping, especially in the rooms, we need a large number of particles (> 10, 000).

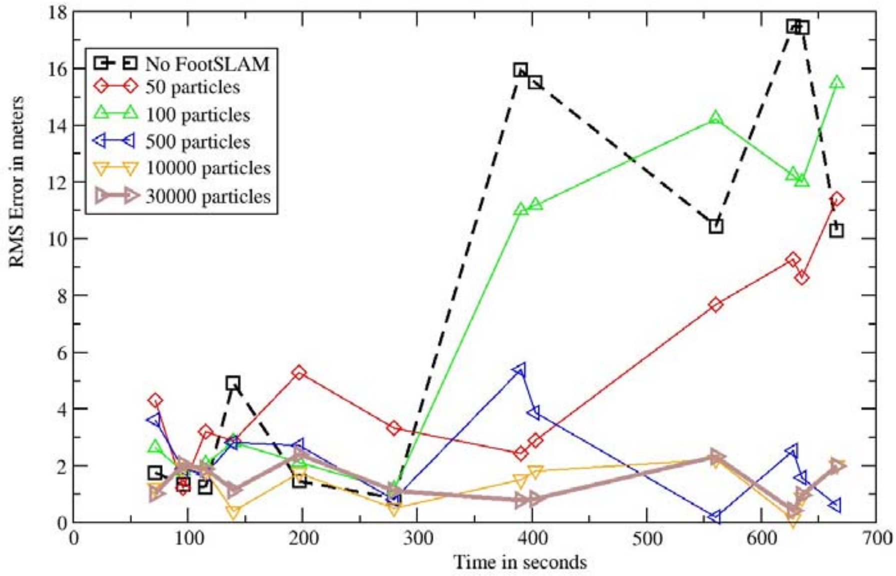


Figure 20: Relative Positioning Accuracy of FootSLAM – indoor with INS only. Each curve is a single run of the particle filter on one data set.

5.5 Principles of PlaceSLAM

To improve on stand-alone FootSLAM we conclude this paper with an overview of an extension - "PlaceSLAM" [22]. It is based on the following idea: If a volunteer, say, performing a mapping walk of a building, were to report their close physical proximity to some reliably detectable "place" to the system, then this would comprise a direct human observation of some aspect of the physical world that is linked to the person's location. To put this into a realistic perspective, let us consider the case where a person starts a walk outside a building and enters through the front entrance. They might choose a spot just inside the door as a suitable "place" and flag their presence when passing this spot. On their subsequent walk through the building, they may remember to indicate their presence when revisiting that same spot. Alternative visual cues could be the entrance to one's own office room, a prominent item such as a fire extinguisher or anything that would allow reliable and repeatable indication of location to within, say, 1 to 2 meters. These observations, as will be shown in this paper, can be used by the map building algorithm to improve accuracy and reliability of the FootSLAM maps, or may aid in a real-time positioning approach. The PlaceSLAM approach is applicable not only to the case of human generated measurements. It pertains also to the situation where RFID tags or other machine-recognizable signals or features are used. For instance, RFID tags might be installed in the environment and the pedestrian might carry an RFID reader. Similarly, the approach could be combined with automatic recognition of prominent visual features by a camera or any other similar sensor. An important contribution of this paper is to distinguish different variants of the approach. First of all, we can distinguish between cases where the true location of these locations is known to the system or not. Secondly, we can look at cases where some unique place identifier (i.e. a name or number) can be reliably associated to an observation or not. In the next section we shall introduce these cases more thoroughly.

5.5.1 Variants of PlaceSLAM

To illustrate the approach we shall use an example. Figure 21 shows an abstracted trajectory of a pedestrian in a certain area. The circles represent places (see definition below) and letters and colors are possible identifiers. On the right side of the figure we see three possible kinds of *placestamp* sequences as input to the estimator: One with perfect association where the placestamps carry the unique letters; one with partial association (colors); and finally the case where only the fact that some place is seen is reported by the user – unknown association.

5.5.1.1 Place with Known Location

The most straightforward case which we can distinguish is human observation of places with known location with perfect association. Here, one or more recognizable places are flagged each time (or just sometimes) when the pedestrian passes them within some small distance. For example, the pedestrian might press a button on their mobile device corresponding to a number posted on a door through which they pass. The location of these places is assumed to be known to the system, at least within a coordinate system relative to the starting position of the pedestrian's current trajectory. Obviously, the easiest such marker is the starting position itself. An abundance of such markers is unrealistic, but one or two markers at known places, such as entrances, could be feasible. For example, such entrances could be reasonably well mapped using GPS from the outside.

5.5.1.2 Unique Place with Unknown Location

Here, one or more recognizable places are flagged each time the pedestrian passes them within some small distance and with a unique signal that allows these places to be differentiated. However, in contrast to above, *we do not assume a known location* of these features. There is one exception: If the user flags their starting point when starting then this will be essentially equivalent to the above case with known location of the feature at the starting point within a relative co-ordinate system. The huge advantage of this scheme is that no prior knowledge of locations is required, so a pedestrian can select suitable places and identifiers themselves. The requirement is that the users can distinguish and flag these places relatively reliably.

5.5.1.3 Places with Unknown Location and Unknown Association

Here, one or more recognizable places are flagged each time the pedestrian passes them within some small distance but *without* any unique signal that allows the identifiers to be differentiated. One can imagine a situation where a pedestrian flags a signal every time they walk through *any* door, or each time they walk past a fire alarm switch, or at other chosen places. The requirement in this case is that the places are not too close together. This is the most difficult yet general case to handle by the system.

At each time step we first perform the standard FootSLAM proposal step and compute its weight update factor for each particle i. Then, if the human has not reported a placestamp we just continue with the normal FootSLAM algorithm. If a placestamp war reported we need to distinguish two cases: 1) If the particle position is further than a predefined threshold distance d_{min} from all previously logged places in the particle's own place map then we assign a new unique identifier to this new place. We then weight the particle with the product of the FootSLAM weight and the PlaceSLAM weight (equation (13) in reference [22]). 2) If the particle is closer than d_{min} from any one previously recorded place then we select the identifier of the place in the particle's place map closest to the particle's current position. We then weight the particle with the product of the FootSLAM weight and the PlaceSLAM weight – equation (12) in [22]. In both cases we update the location of the place according to its previous location distribution and to the location of the particle.

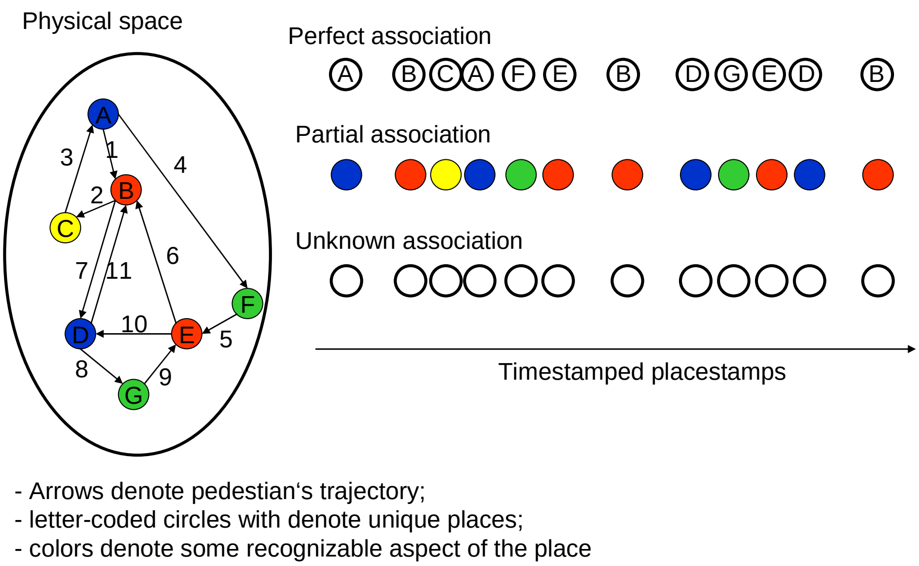


Figure 21: Example of a sequence of places visited during a pedestrian's walk and different resulting sequences of placestamps.

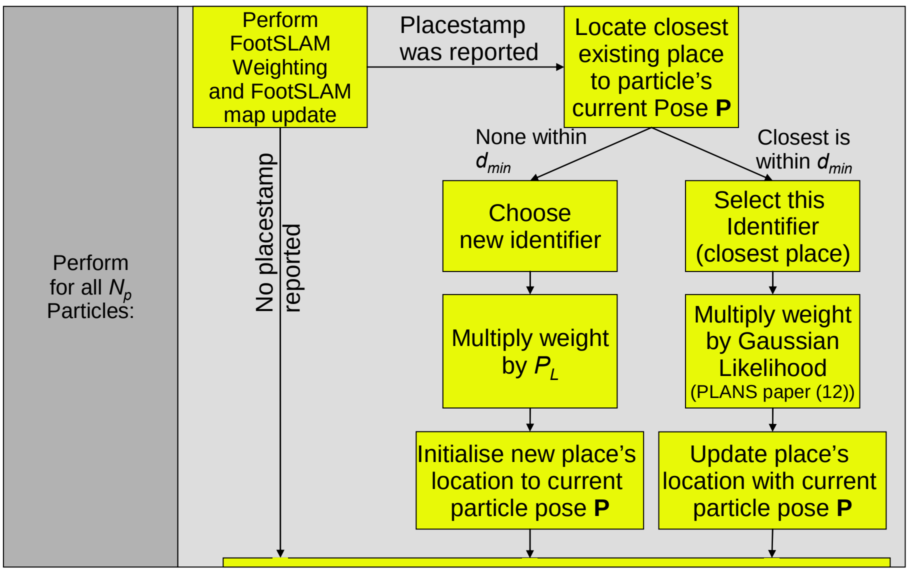


Figure 22: Summary of the PlaceSLAM algorithm for the case of unknown association. For details and definition of d_{min} and P_L see the PLANS paper [22].

5.5.2 Practical Considerations

Practical applications of PlaceSLAM require the pedestrian to cooperate and to "think along". In the simplest case (for the user) a binary signal is needed such as pressing a button on a mobile device. It is only suited when the involved pedestrians are aware of the mapping process and when places are correctly chosen.

5.5.3 PlaceSLAM and FootSLAM Combination: Results

Experimental evaluation of PlaceSLAM so far indicates that its addition to FootSLAM can increase the convergence speed and can improve the accuracy of the maps, especially when using few particles in the particle filter. Figure 23 shows two maps generated with three places in each case. The left example is an outdoor-indoor-outdoor situation where the building was the ground floor of the one used in Figure 19. Figure 24 shows the evolution of the error for the outdoor-indoor-outdoor situation example over time. We clearly see how PlaceSLAM converges more quickly which can be very valuable in a rapid collaborative mapping scenario.

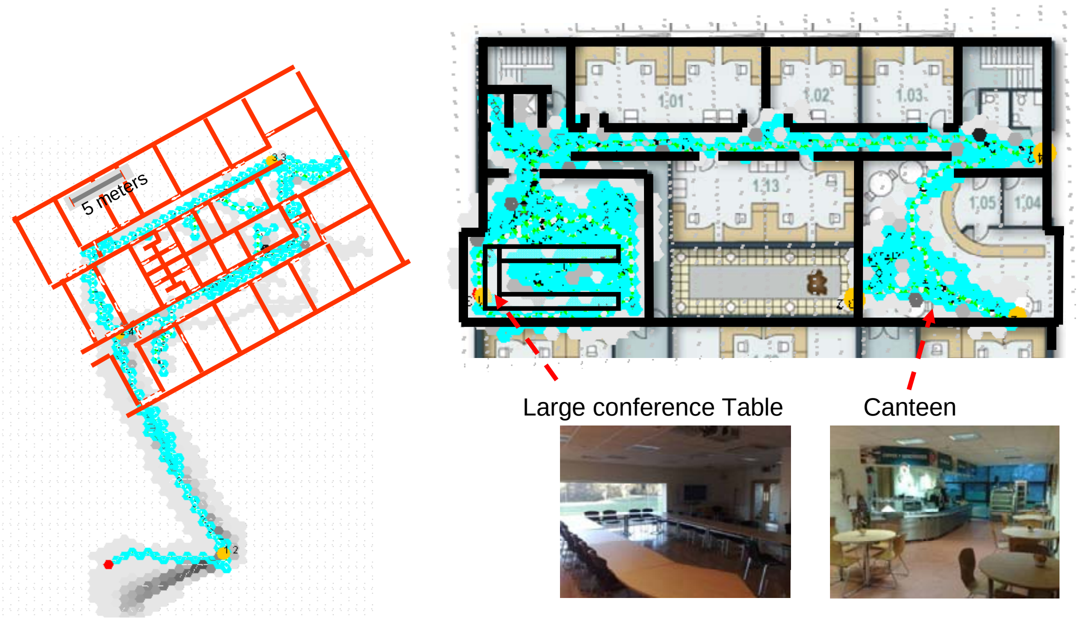


Figure 23: Two different examples of FootSLAM (blue) and PlaceSLAM (yellow circles) map. In both cases there were three human-reported places with unknown association. The maps were rotated and scaled to fit the true layout for comparison.

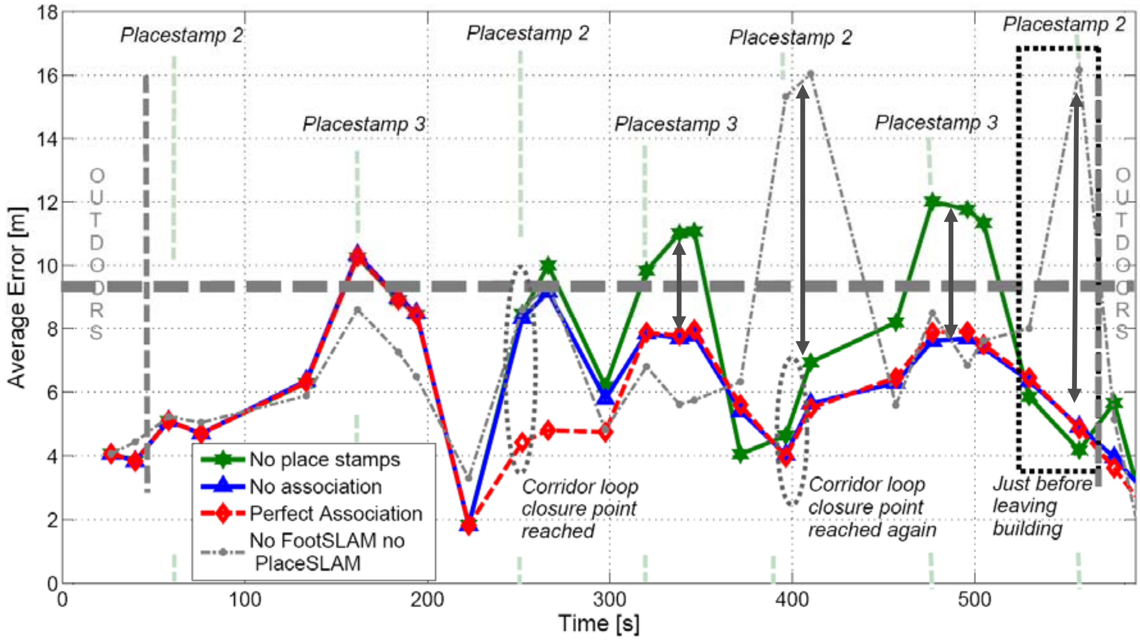


Figure 24: Average position error of the particle filter over time as a result evaluating 100 independently seeded runs of a particle filter with 50000 particles on the same data set. Note how using placestamps with perfect association yields a smaller error immediately after the first loop closure, whereas unknown association takes a little longer to achieve the same level of performance. The green curve corresponds to using FootSLAM alone.

6.0 CONCLUSIONS AND OUTLOOK

We have discussed the inner workings of FootSLAM, a new pedestrian localization technique that uses the principles of simultaneous localization and mapping, originally developed for robots, for assisting human users. FootSLAM, in its current implementation 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, FootSLAM does not require specific sensors, such as cameras or laser scanners for detecting features. Nevertheless, FootSLAM succeeds in preventing an unbounded growth of the error over time, which previously was unavoidable when inertial sensors were used and no a priori map knowledge was available. Additionally, building maps, which are learnt automatically from measurements generated by pedestrians that roam a building, are generated and can be used for map-based pedestrian dead reckoning.

FootSLAM, works robustly in an offline mode on real data measured with low-cost foot-mounted IMUs. The generated maps are sufficiently accurate to facilitate map-based pedestrian dead reckoning.

Wherever feasible, FootSLAM can be improved by applying PlaceSLAM. While PlaceSLAM puts a small additional burden on the user or requires additional sensors to re-identify previously visited places, it further improves robustness, reduces computational complexity and increases accuracy.

In its current form, FootSLAM is a promising scheme that has been shown to work well in the semi-controlled environments we have tested it in, so far. In our future work we will strive to test it under increasingly realistic and challenging scenarios. Major challenges will involve its application in large compounds or structures, such as airports. Combining the measurements of a large set of users in an offline, yet efficient manner is important for its large scale application. From our current perspective, we see the most interesting challenge in facilitating its application to provide real-time mapping and mutual position-awareness to collaborating users in scenarios involving emergency or law enforcement operations.

REFERENCES

[1] Ruizhi Chen, Wei Chen, Xiang Chen, Xu Zhang, and Yuwei Chen "Sensing strides using EMG signal for pedestrian navigation", GPS Solutions, Springer 2010, DOI: 10.1007/s10291-010-0180-x.

[2] E. Foxlin, "Pedestrian tracking with shoe-mounted inertial sensors," IEEE Computer Graphics and Applications, Vol. 25, no. 6, November 2005.

[3] Y. BarShalom, X. R. Li and T. Kirubarajan, "Estimation with Applications to Tracking and Navigation: Algorithms and Software for Information Extraction", J. Wiley and Sons, 2001.

[4] Stephane Beauregard and Harald Haas, "Pedestrian Dead Reckoning: A Basis for Personal Positioning", Proceedings of the 3rd Workshop on Positioning, Navigation and Communication (WPNC'06), 2006.

[5] Z. Liu and C-H. Won , "Knee and Waist Attached Gyroscopes for Personal Navigation: Comparison of Knee, Waist and Foot Attached Inertial Sensors", Proceedings of IEEE/ION PLANS 2010, May 4-6, 2010.

[6] Michael Angermann, Patrick Robertson, Thomas Kemptner, and Mohammed Khider, "A High Precision Reference Data Set for Pedestrian Navigation using Foot-Mounted Inertial Sensors", International Conference on Indoor Positioning and Indoor Navigation (IPIN), 15-17 September 2010, Zürich, Switzerland, 2010.

[7] 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.

[8] M. Khider, S. Kaiser, P. Robertson, M. Angermann, "The Effect of Maps-Enhanced Novel Movement Models on Pedestrian Navigation Performance", European Navigation Conference (ENC-GNSS 2008), Toulouse, France, April 2008.

[9] 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.

[10] S. Thrun, W. Burgard, and D. Fox, "Probabilistic Robotics", MIT-Press, 2005.

[11] 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.

[12] 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, February 2002.

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

[14] E. Foxlin and S. Wan, "Improved pedestrian navigation based on drift-reduced MEMS IMU chip," in Proc. of the 2010 ION International Technical Meeting, San Diego, California, USA, January 2010.

[15] 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 Proc. European Navigation Conference (ENC-GNSS 2009), Naples, Italy, May 2009.

[16] I. Skog, J.-O. Nilsson, P. Händel, "Evaluation of Zero-Velocity Detectors for Foot-Mounted Inertial Navigation Systems", in Proc. IPIN 2010, Zurich, Switzerland, September 2010.

[17] "Reference Data Sets for Multisensor Pedestrian Navigation", http://www.kn-s.dlr.de/indoornav/refdataset.html.

[18] I. Skog, P. Händel, J. Rantakokko, and J.-O. Nilsson, "Zero-velocity detection - An algorithm evaluation", to appear in IEEE Trans. on Biomedical Engineering, Fall 2010.

[19] H. Durrant-Whyte, T. Bailey, "Simultaneous Localisation and Mapping (SLAM): Part I The Essential Algorithms", Robotics and Automation Magazine, June 2006.

[20] P. Robertson, M. Angermann, and B. Krach, "Simultaneous Localization and Mapping for Pedestrians using only Foot-Mounted Inertial Sensors", in Proc. Ubicomp 2009, Orlando, Florida, USA, 30 September – 3 October 2009.

[21] P. Robertson, M. Angermann, B. Krach, and M. Khider, "Inertial Systems Based Joint Mapping and Positioning for Pedestrian Navigation", Proc. ION GNSS 2009, Savannah, Georgia, USA, 22 September – 25 September 2009.

[22] P. Robertson, M. Angermann and M. Khider, "Improving Simultaneous Localization and Mapping for Pedestrian Navigation and Automatic Mapping of Buildings by using Online Human-Based Feature Labeling", in Proc. IEEE/ION PLANS 2010, Palm Springs, CA, USA, May 2010.

[23] Xsense website, http://www.xsens.com/.

[24] Vicon Bonita product website, http://www.vicon.com/products/bonita.html.

[25] "FootSLAM and PlaceSLAM videos", http://www.kn-s.dlr.de/indoornav/footslam_video.html.