

Article

A Particle Filter for Smartphone-Based Indoor Pedestrian Navigation

Andrea Masiero *, Alberto Guarnieri †, Francesco Pirotti † and Antonio Vettore †

CIRGEO (Interdepartmental Research Center of Geomatics), University of Padova, via dell'Università 16, 35020 Legnaro (PD), Italy; E-Mails: alberto.guarnieri@unipd.it (A.G.); francesco.pirotti@unipd.it (F.P.); antonio.vettore@unipd.it (A.V.)

† These authors contributed equally to this work.

* Author to whom correspondence should be addressed; E-Mail: masiero@dei.unipd.it; Tel.: +39-320-026-9583.

External Editors: Naser El-Sheimy and Aboelmagd Noureldin

Received: 4 September 2014; in revised form: 13 October 2014 / Accepted: 24 October 2014 / Published: 5 November 2014

Abstract: This paper considers the problem of indoor navigation by means of low-cost mobile devices. The required accuracy, the low reliability of low-cost sensor measurements and the typical unavailability of the GPS signal make indoor navigation a challenging problem. In this paper, a particle filtering approach is presented in order to obtain good navigation performance in an indoor environment: the proposed method is based on the integration of information provided by the inertial navigation system measurements, the radio signal strength of a standard wireless network and of the geometrical information of the building. In order to make the system as simple as possible from the user's point of view, sensors are assumed to be uncalibrated at the beginning of the navigation, and an auto-calibration procedure of the magnetic sensor is performed to improve the system performance: the proposed calibration procedure is performed during regular user's motion (no specific work is required). The navigation accuracy achievable with the proposed method and the results of the auto-calibration procedure are evaluated by means of a set of tests carried out in a university building.

Keywords: indoor navigation; positioning; sensor fusion; nonlinear filtering; smartphones geolocation

1. Introduction

Thanks to certain socially relevant applications (e.g., localizing and tracking people inside buildings during emergencies), indoor navigation is recently becoming a topic of wide interest among the research community. Despite the recent developments in localization with wireless sensor networks (e.g., by means of the radio signal strength (RSS), [1–7]) and in pedestrian tracking (e.g., urban dead-reckoning, [8,9]), indoor navigation is still considered an open problem [10–13]. Specifically, three challenging aspects can be recognized in indoor navigation: the required accuracy is typically higher than in outdoor applications, the GPS signal is typically not available and there is low reliability of the sensor measurements (e.g., inertial navigation system (INS) measurements and RSS).

The main issue in the use of INS measurements to provide updates of the estimated position is that such an estimate quickly drifts from the real one. On the other hand, position estimation based on RSS measurements does not suffer from drifts; however, because of environment variability and signal instability, the estimation error of such an approach is typically quite large.

Since the use of none of the considered approaches (INS and RSS) can allow by itself obtaining a sufficiently good estimation error in the considered conditions of interest, a commonly adopted strategy to tackle this problem relies on the integration between the data collected by several types of sensors to achieve more robust localization results [14–17]. In particular, in this paper, an indoor navigation system with minimal positioning sensor equipment is considered: the goal of this work is to enable navigation with low-cost mobile devices (typically carried by the user's hand) in indoor and other critical environments, e.g., the proposed navigation algorithm can be executed on a smartphone, which estimates its own position inside a building by combining the information collected from the Wi-Fi network (RSS) with measurements derived from the embedded INS sensor (the smartphone considered here is provided with a three-axis accelerometer and three-axis magnetometer). Furthermore, a map of the building is assumed to be available to the tracking algorithm: this information can be either provided by the Wi-Fi network or acquired by image plots or plans of the building.

Similarly to other indoor navigation methods recently proposed in the literature [17,18], in this work, the tracking problem is faced from a probabilistic point of view: the probability distribution of the current position is described as a sum of particles whose movements are determined by the sensor measurements (see [19–21] for a description of particle filters and [22–25] for some particle filtering applications).

The approach adopted in this paper is actually a generalization of the particle filter proposed in [17]: the proposed changes aim at compensating measurement errors due to the use of uncalibrated sensors. Indeed, in order to make the use of the system as simple as possible to the user, sensors are not calibrated by using dedicated specific procedures. Instead, the particle filter proposed in [17] is properly modified to deal with measurements from uncalibrated sensors. Then, an auto-calibration procedure is adopted for the magnetic sensor to improve the system performance. Differently from calibration algorithms typically used in similar systems [26–28], this calibration procedure does not require any specific action by the user: it is performed during the navigation, without affecting the user's regular moves in any way.

This way, the adopted navigation procedure results in a new algorithm that integrates the positional information derived from INS and Wi-Fi RSS measurements with the geometrical information provided by the building map, while dealing with uncalibrated sensors, as well.

The considered method has shown a significant improvement in the navigation accuracy with respect to the particle filter proposed in [17] on some experiments carried out in a university building using the standard Wi-Fi network.

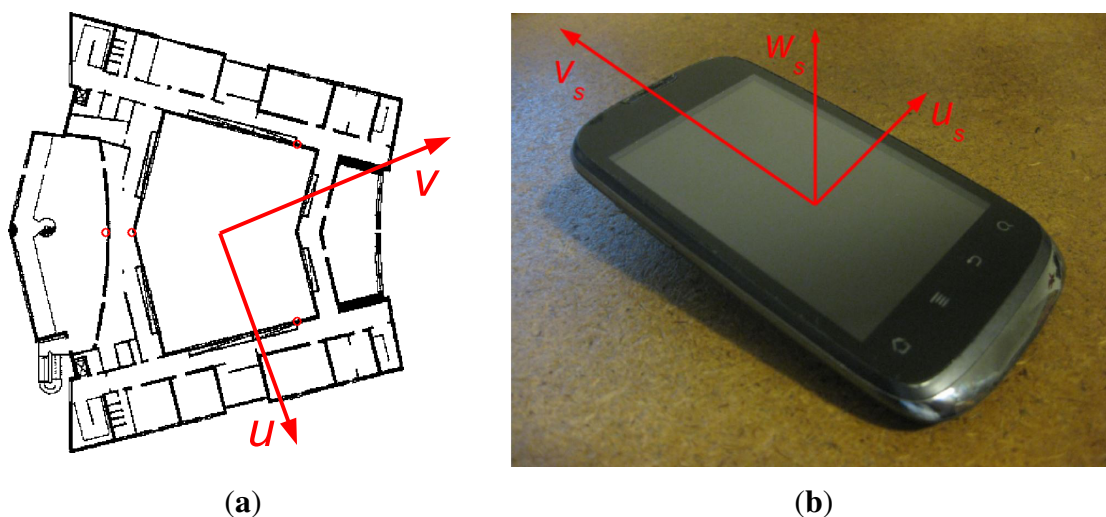
The paper is organized as follows: Section 2 provides a basic description of the navigation system and its relation to [17] and other previous works. Section 3 introduces the tracking and sensor data fusion approach used in this work. Section 4 presents the procedure adopted to perform the auto-calibration of the magnetometer. Finally, in Section 5, the performance of the method is experimentally validated.

2. System Description and Relation to Previous Works

Thanks to the technological developments in electronics and MEMS-based sensors [29–32], nowadays, most mobile devices (e.g., smartphones and tablets) are provided with several embedded sensors. Hence, differently from other works, which considered the use of external sensors (e.g., [8,9,17]), here, an indoor navigation system based on the use of the sensors embedded in standard mobile devices is considered.

To be more specific, a minimal equipment of embedded sensors is required: a three-axis accelerometer, a three-axis magnetometer (e.g., MEMS-based inertial navigation system) and a Wi-Fi antenna. Furthermore, the navigation system is assumed to have access to the standard Wi-Fi network available inside the building. The installation of additional access points (APs) is not required (access points on one of floors of the considered building are shown as red circles in Figure 1a).

Figure 1. (a) Coordinate system in the university building used as a test field, and (b) the smartphone coordinate system (u_s, v_s, w_s) . Smartphone used for the test: Huawei Sonic U8650.



In addition, to make the use of the system as simple as possible from the user's point of view, no specific initial sensor calibration procedure is needed: the tracking algorithm that will be presented in Section 3 is designed to deal with uncalibrated sensors. Furthermore, as shown in Section 4, in order to improve the navigation performance, the system is designed to auto-calibrate the magnetic sensor during the navigation, e.g., exploiting regular user's movements.

The rationale is that of using a dead reckoning-like approach [8,9]: detect the human steps by means of a proper analysis of the accelerometer measurements [33], then use the combination of magnetometer and accelerometer measurements to estimate the movement direction with respect to the north [28].

Since, in terrestrial applications, usually, the height with respect to the floor is of minor interest, then, hereafter, navigation is considered as a planar tracking problem (exceptions to this assumption are admitted, e.g., to deal with stairs and lifts). Let (u_t, v_t) be the position of the device (e.g., smartphone), expressed with respect to the north and east directions (*i.e.*, global reference system; Figure 1a shows the coordinate system in a university building used as the test field), before the t -th step; then:

$$\begin{bmatrix} u_{t+1} \\ v_{t+1} \end{bmatrix} = \begin{bmatrix} u_t \\ v_t \end{bmatrix} + s_t \begin{bmatrix} \sin \alpha_t \\ \cos \alpha_t \end{bmatrix} \quad (1)$$

where s_t is the length of the t -th step and α_t is the corresponding heading direction. An estimation of the initial position (u_0, v_0) is assumed to be (*a priori*) available.

The step length s_t is assumed to be provided by a proper analysis of the accelerator measurements [33]. Alternatively, s_t can be fixed to a constant value (an approximation of the mean step length): the tracking algorithm described in Section 3 is designed to compensate (relatively small) step length errors.

In most of the cases of common interest, the mobile device is supposed to be carried by the user's hand (this is the case considered in the simulations of Section 5); however, the navigation system is expected to work in different conditions, as well. To be more specific, the system is expected to work as long as the heading direction is approximately fixed with respect to the local coordinate system (more details on this assumption are provided in the Appendix) (u_s, v_s, w_s) (the local coordinate system for a smartphone used in the simulations of Section 5 is shown in Figure 1b). However, small deviations are allowed: the system of Section 3 is designed to estimate and correct heading direction discrepancies, with an absolute value lower than 36 degrees, with respect to the conventional direction.

Without loss of generalization, hereafter, the v_s and the w_s axes will be assumed to approximately correspond with the heading direction and with the vertical direction, respectively. Notice that these assumptions are not requirements of the proposed approach, but will ease the presentation of the method and the analysis of the calibration results shown in Section 5.

Notice that the update of Equation (1) according to the sensor measurements provides an approximation of the device trajectory. However, it is well known that, because of measurement errors, the trajectory estimated by means of Equation (1) quickly drifts from the correct one. In order to make the estimation method more robust, information provided by the dead reckoning-like approach has to be integrated with that provided by RSS measurements and the geometrical characteristics of the building (Section 3). RSS measurements of the standard Wi-Fi networks are provided by the corresponding antenna in the smartphone. The geometrical characteristics of the building are assumed to be pre-loaded on the navigation device or downloaded by means of the Wi-Fi network at the beginning of the navigation. As the mobile device gets inside of the building, it starts communicating with the Wi-Fi network, hence the device is considered as immediately ready to go (the total time for connecting the device to the network and to receive the information required by the navigation system depends

on the specific conditions of the Wi-Fi network (e.g., signal strength, connection data rate, number of users connected to the network); however, the communication step is typically very fast (e.g., quantity of information to be passed < 100 kBytes in our simulations)).

Several nonlinear filtering strategies have been previously considered in the literature for the integration of information (e.g., INS and RSS measurements) in pedestrian indoor navigation [8,9,16,17]. Among such possible strategies, a particle filter approach [19] is considered. In particular, our approach can be considered as a generalization of the particle filter proposed in [17,34], and that will be briefly summarized in the following (the reader is referred to [17,34] for a detailed description).

2.1. Widyawan et al. Particle Filter

Particle filter techniques are statistical methods widely used in complex (possibly nonlinear) estimation problems [19–24]. The variable of interest at time t , q_t (e.g., the device position in the navigation problem considered here), is described by means of a density function $p(q_t|Y_t)$, which depends on the values of the past measurements Y_t : the goal of the particle filter is to properly estimate such posterior density $p(q_t|Y_t)$ (and update it when new measurements are available). Particle filters are particularly effective when dealing with complex systems, because of the two following characteristics:

- The probability function $p(q_t|Y_t)$ is not restricted to be Gaussian nor univariate; instead, it is described as the sum of a set of n particles.
- No restriction is imposed on the dynamics of the system (*i.e.*, it can deal with both linear and nonlinear systems).

However, when dealing with complex systems (e.g., nonlinear and with non-univariate density $p(q_t|Y_t)$), a large number of particles n has to be used in order to ensure good filtering performance: hence, the ability to deal with difficult scenarios is paid for with an increase of the computational complexity of the filter.

Several particle filter methods have been proposed in the literature, and each of them can be summarized as a specific algorithm used in order to properly updated the particle description of $p(q_t|Y_t)$ when new measurements are available. The rest of this section is dedicated to the description of the particle filter for the indoor navigation proposed in [17,34], while a generalization of such an approach will be presented in Section 3 in order to efficiently deal with uncalibrated sensors (while keeping the number of particles n relatively small).

Let y_t be the vector of measurements corresponding to the t -th step and Y_t be the collection of measurements $\{y_\tau\}$ from $\tau = 0$ to $t - 1$. Furthermore, let $q_t = [u_t \ v_t]^T$; then the probability distribution of estimated position q_t after the $(t - 1)$ -th step is expressed as follows:

$$p(q_t|Y_t) = \sum_{i=1}^n w_{i,t} \delta(q_t - q_{i,t}) \quad (2)$$

where $q_{i,t}$ and $w_{i,t}$ are the position and the weight of the i -th particle at t , while $\delta(\cdot)$ is the Dirac delta function.

In order to simplify the presentation, with a slight abuse of notation, hereafter, the subscript index t will be referred to as a time index, as well.

In this subsection, the INS sensor measurements are assumed to be unbiased, but affected by zero-mean random Gaussian noise, with heading direction and step length variance σ_α^2 and σ_s^2 , respectively. Then, the position of the i -th particle at time $t + 1$ is updated similarly to Equation (1):

$$\hat{q}_{i,t+1} = q_{i,t} + s_{i,t} \begin{bmatrix} \sin \alpha_{i,t} \\ \cos \alpha_{i,t} \end{bmatrix} \tag{3}$$

where $\alpha_{i,t}$ and $s_{i,t}$ are randomly sampled from the proper Gaussian distributions: $\alpha_{i,t} \sim \mathcal{N}(\alpha_t, \sigma_\alpha^2)$, $s_{i,t} \sim \mathcal{N}(s_t, \sigma_s^2)$. σ_α and σ_s are set to proper constant values ($\sigma_\alpha = \pi/2$, $\sigma_s = 0.15$ m in [17]).

Exploiting an RSS channel model [3], the RSS measurements are converted to distance measurements. Typically, the error on each measured distance is assumed to be normally distributed [3,17]. Then, the comparison of measured distances with the distances from the access points to the estimated device position yields the RSS probability $p_{RSS}(\hat{q}_{i,t+1}|y_t)$, that is the probability of being positioned in $\hat{q}_{i,t+1}$ given the RSS measurement in y_t (see [3,17] for details on $p_{RSS}(\cdot)$).

Then, the particle weights are updated as follows:

$$\hat{w}_{i,t+1} = \begin{cases} p_{RSS}(\hat{q}_{i,t+1}|y_t) & \text{if } y_t \text{ contains an RSS measurement} \\ 1/n & \text{otherwise,} \end{cases} \tag{4}$$

Furthermore, particles with trajectories that violate the geometric constraints of the building are discarded. Then, the weights $\{\hat{w}_{i,t+1}\}_{i=1,\dots,n}$ are scaled in order to be normalized to one, *i.e.*, $\sum_{i=1,\dots,n} \hat{w}_{i,t+1} = 1$, and the following distribution for the particles at time $t + 1$ is obtained:

$$p(q_{t+1}|Y_{t+1}) \approx \sum_{i=1}^n \hat{w}_{i,t+1} \delta(q_{t+1} - \hat{q}_{i,t+1}) \tag{5}$$

In the above equation, part of the particles may have very low weights (zero if the particle trajectory violates the building constraints); then, the effective number \hat{n} of particles in Equation (5) is typically lower than n (a detailed study of the effective number of particles goes beyond the goal of this paper; for simplicity of exposition, hereafter, \hat{n} at time t will be considered as the number of particles, such that $w_{i,t}$ is significantly greater than zero (*i.e.*, larger than a given threshold); the reader is referred to [20,21] for a more detailed description of particle filter characteristics). In order to preserve the effectiveness of the particle representation, the final approximation of the position distribution at time $t + 1$ is obtained by resampling n particles from Equation (5), *i.e.*, the following actions has to be done for each resampled particle i' : an index i is randomly sampled accordingly to the probability $\hat{w}_{i,t+1}$; then, $q_{i',t+1}$ is sampled from a Gaussian distribution centered at $\hat{q}_{i,t+1}$, and its weight is set to $w_{i',t+1} = 1/n$ (notice that the resampling step increases the estimation variance; see [35] for a comparison of different resampling schemes).

The above approach can be summarized with the following procedure to be executed each time a new measurement is available:

- (1) For each particle i :
 - (1.a) draw a sample $\hat{q}_{i,t+1}$ from the proposal distribution (*i.e.*, draw samples $\alpha_{i,t}$ and $s_{i,t}$ from $\mathcal{N}(\alpha_t, \sigma_\alpha^2)$ and $\mathcal{N}(s_t, \sigma_s^2)$, respectively, then compute $\hat{q}_{i,t+1}$ with Equation (3)).

- (1.b) Set $\hat{w}_{i,t+1} = 0$ if $\hat{q}_{i,t+1}$ violates building geometrical constraints; otherwise, set the weight $\hat{w}_{i,t+1}$ as in Equation (4).
- (2) Scale the particle weights in order to normalize their sum to one: $\sum_{i=1,\dots,n} \hat{w}_{i,t+1} = 1$.
- (3) Resample n particles from Equation (5) and set $w_{i',t+1} = 1/n, \forall i'$.

Notice that the number of operations to be executed for each particle in the above procedure can be upper bounded by a proper constant (and lower bounded, as well); hence, the computational complexity of the above particle filter is linear with respect to the number of particles n .

The reader is referred to [17,34] for details on the implementation of this particle filtering approach.

2.2. Issues Related with the Use of Uncalibrated Sensors

The particle filter summarized in the previous subsection has been proposed in [17,34] for devices with calibrated sensors. In such conditions, measurement errors can be assumed to be mostly random: in this case, the random sampling of the heading direction, $\alpha_{i,t} \sim \mathcal{N}(\alpha_t, \sigma_\alpha^2)$, and of the step length, $s_{i,t} \sim \mathcal{N}(s_t, \sigma_s^2)$, and the filtering obtained by means of imposing the geometrical constraints of the building (Step 1.b) of the procedure presented in the previous subsection) typically allow one to properly deal with these errors.

However, when working with uncalibrated sensors, measurement errors are typically correlated and larger (in absolute value) with respect to the calibrated case. This makes much more challenging the particle filter tracking task: this often leads to a significant decrease of \hat{n} and, if n is not sufficiently large, to a tracking fault of the particle filter, *i.e.*, none of the particles describe adequately the device position; hence, the particle filter loses the correct track (this case occurred quite often in our simulations; in order to reduce the occurrence of such case, then, in our simulations of the particle filter described in Subsection 2.1, Step 1.a has been repeated (for a maximum of five times) when the generated sample $\hat{q}_{i,t+1}$ violates building geometrical constraints at Step 1.b).

It is well known that using a sufficiently large number of particles n , particle filtering techniques allow one to approximate arbitrarily well the considered system. However, the considered tracking algorithm is supposed to run in real time on a mobile device with limited resources (*i.e.*, computational power and available energy); hence, the number of particles n should be as small as possible in order to reduce the computational load and the power consumption of the device.

The above considerations motivate the generalization that will be presented in the following section of the particle filter of Subsection 2.1 in order to improve its performance (e.g., allow one to obtain similar results with a smaller number of particles) in the uncalibrated sensors case.

3. Tracking with Uncalibrated Sensors

The use of uncalibrated sensors can introduce temporally correlated measurement errors for both the heading direction and the step length. The following two subsections will separately consider these two cases, while Subsection 3.3 will summarize the overall approach.

3.1. Heading Direction Correction

The heading direction measurement α_t can be statistically described as follows:

$$\alpha_t = \bar{\alpha}_t + \alpha_{bs,t} + \alpha_{be,t} + e_{\alpha,t} \quad (6)$$

where $\bar{\alpha}_t$ is the correct heading direction, $\alpha_{bs,t}$ is an angular bias due to the use of an uncalibrated magnetometer, $\alpha_{be,t}$ is an angular bias due to local perturbations of the terrestrial magnetic field typically due to the presence of ferromagnetic materials in the environment and $e_{\alpha,t}$ is a temporally random measurement error (that can typically be assumed to be white and Gaussian). The goal of this subsection is to approximately compensate for the bias $\alpha_{b,t} = \alpha_{bs,t} + \alpha_{be,t}$.

The following considerations are now in order:

- When the device moves along a rectilinear trajectory, *i.e.*, $\bar{\alpha}_t$ is (approximately) fixed to a constant value for $t = t_s, \dots, t_e$, then $\alpha_{bs,t} \approx \alpha_{bs,t'}$ for $t_s \leq t \leq t_e, t_s \leq t' \leq t_e$.
- Assuming the magnetic field to be sufficiently smooth in the considered environment, then locally $\alpha_{be,t} \approx \alpha_{be,t'}$ for $t \approx t'$.
- Human trajectories are usually quite regular.

Then, the rationale is that of decomposing the device motion in piecewise (approximately) rectilinear trajectories. Each piece of such trajectories being approximately rectilinear and typically not so long, then:

$$(\alpha_{bs,t} + \alpha_{be,t}) \approx (\alpha_{bs,t'} + \alpha_{be,t'}) \quad (7)$$

for t and t' in the same piece of (approximately) rectilinear trajectory.

The end of a piece of the (approximately rectilinear) trajectory (and consequently, the beginning of a new piece) is detected when the following condition holds:

$$|\alpha_t - \alpha_{t-1}| > \alpha_{thr} \vee |\alpha_t - \alpha_{t-2}| > \alpha_{thr} \vee |\alpha_t - \alpha_{t-3}| > \alpha_{thr} \quad (8)$$

where \vee indicates the logic operator “or” and α_{thr} is a proper threshold ($\alpha_{thr} = \pi/6$ in the simulations of Section 5).

Accordingly to Condition (8), the trajectory is divided in non-overlapping sections of (approximately) rectilinear trajectories. Let $t_{s,j}$ and $t_{e,j}$ be the initial and ending time instant for the j -th (approximately rectilinear) section. Then, in order to compensate for the bias $\alpha_{b,t}, t_{s,j} \leq t \leq t_{e,j}$, Step 1.b of the particle filtering procedure presented in Subsection 2.1 is modified as follows:

- (1.b.1) If $\hat{q}_{i,t+1}$ violates the geometrical constraints of the building, then resample $\hat{q}_{i,t+1}$ as follows: draw a rotation angle sample $\hat{\alpha}_{b,t}$ from $\mathcal{U}(-\alpha_{\max}, \alpha_{\max})$, and rotate $(\hat{q}_{i,t_{s,j}}, \dots, \hat{q}_{i,t_{e,j}})$ of an angle $\hat{\alpha}_{b,t}$ around the vertical axis passing through $\hat{q}_{i,t_{e,j-1}}$.
- (1.b.2) Set $\hat{w}_{i,t+1} = 0$ if $\hat{q}_{i,t+1}$ violates building geometrical constraints; otherwise, set the weight $\hat{w}_{i,t+1}$ as in Equation (4).

Since the angular bias $\alpha_{b,t}$ is approximately constant over an (approximately rectilinear) section of the trajectory $t_{s,j} \leq t \leq t_{e,j}$, then Step 1.b.1 aims at compensating for such angular bias (if present),

subtracting a randomly sampled angular bias $\hat{\alpha}_{b,t}$ to the heading direction measurements over all of the trajectory section $t_{s,j} \leq t \leq t_{e,j}$. Since no information about such an angular bias is available, $\hat{\alpha}_{b,t}$ is randomly sampled from a uniform distribution in the interval $(-\alpha_{\max}, \alpha_{\max})$, *i.e.*, $\mathcal{U}(-\alpha_{\max}, \alpha_{\max})$, where $\alpha_{\max} = \pi/5$ in our simulations of Section 5.

Then, Step 1.b.2, which corresponds to Step 1.b in Subsection 2.1, has the task of properly filtering out particles associated with improper values of $\hat{\alpha}_{b,t}$ (and of the other trajectory parameters).

3.2. Step Length Correction

The use of an uncalibrated accelerometer leads to a possible bias in the step length measurements: $s_t = \bar{s}_t + s_{b,t} + e_{s,t}$, where s_t is the measured step length, \bar{s}_t is the correct value of the step length, $s_{b,t}$ is the bias and $e_{s,t}$ is a zero-mean white Gaussian noise with variance σ_s^2 .

In order to maximize the flexibility of the filter, but without excessively affecting the complexity of the filtering procedure, analogous to the previous Subsection, it is assumed that the value of the step length bias can be different in different sections of the trajectory; however, it is assumed (approximately) constant within each section (for simplicity of the filtering procedure, the same trajectory sections determined in Subsection 3.1 are considered here; nevertheless, the determination of a possible relation between the characteristics of human walking and of the step length bias will be the subject of a future investigation), *i.e.*, $s_{b,t} \approx s_{b,t'}$ for $t_{s,j} \leq t \leq t_{e,j}$, $t_{s,j} \leq t' \leq t_{e,j}$.

Then, Step 1.a of Subsection 2.1 is modified as follows in order to take into account a possible step length bias:

(1.a) Draw a sample $\hat{q}_{i,t+1}$ from the proposal distribution: draw samples $\alpha_{i,t}$ and $s_{i,t}$ from $\mathcal{N}(\alpha_t, \sigma_\alpha^2)$ and $\mathcal{N}(s_t + \hat{s}_{b,i,t}, \sigma_s^2)$, respectively; then, compute $\hat{q}_{i,t+1}$ with Equation (3).

where, for each particle i , $\hat{s}_{b,i,t}$ is randomly drawn from a uniform distribution $\mathcal{U}(-s_{\max}, s_{\max})$ at the beginning of each section of the trajectory. $\hat{s}_{b,i,t}$ is fixed to such a value for all of the time instants of the same section of the trajectory, *i.e.*, $\hat{s}_{b,i,t} = \hat{s}_{b,i,t'}$ for $t_{s,j} \leq t \leq t_{e,j}$, $t_{s,j} \leq t' \leq t_{e,j}$.

Then, as previously shown, Step 1.b.2 has the task of properly filtering out particles associated with improper values of $\hat{s}_{b,i,t}$ (and of the other trajectory parameters).

Finally, it is worth noticing that, when only INS measurements are available, the particle filter of Subsection 2.1 in open space results in being a biased estimator. For simplicity of exposition, assume that both heading direction and step length measurements are unbiased (or have already been properly compensated for). Furthermore, without loss of generality, it is assumed that $\bar{\alpha}_t = 0$ and that the device moves at time t from the origin of the global coordinate system, *i.e.*, $q_t = [0 \ 0]^T$.

Then, the following condition holds for an unbiased estimator:

$$\mathcal{E}[q_{t+1}|Y_{t+1}] = \begin{bmatrix} 0 \\ \bar{s}_t \end{bmatrix} \tag{9}$$

where $\mathcal{E}[\cdot]$ indicates the expectation operator.

Since the α_t distribution is symmetric with respect to zero, it is simple to show that $\mathcal{E}[q_{t+1}|Y_{t+1}] = \begin{bmatrix} 0 \\ s \end{bmatrix}$ for a certain value of s . In order to compute s , consider the Taylor second order approximation of $\mathcal{E}[s]$ for $\alpha_t = e_{\alpha,t} \approx 0$ and $s_t = \bar{s}_t + e_{s,t} \approx \bar{s}_t$. Then,

$$\begin{aligned} \mathcal{E}[s_t \cos(\alpha_t)] &\approx \bar{s}_t + [\cos(0) - \bar{s}_t \sin(0)] \begin{bmatrix} \mathcal{E}[e_{s,t}] \\ \mathcal{E}[e_{\alpha,t}] \end{bmatrix} + \frac{1}{2} [-\sin(0) - \bar{s}_t \cos(0)] \begin{bmatrix} \mathcal{E}[e_{s,t}^2] \\ \mathcal{E}[e_{\alpha,t}^2] \end{bmatrix} \quad (10) \\ &\approx \bar{s}_t + 0 + \frac{1}{2} \begin{bmatrix} 0 & -\bar{s}_t \end{bmatrix} \begin{bmatrix} \sigma_s^2 \\ \sigma_\alpha^2 \end{bmatrix} = \bar{s}_t \left(1 - \frac{\sigma_\alpha^2}{2}\right) \quad (11) \end{aligned}$$

Thus, for $\sigma_\alpha^2 > 0$, then $\mathcal{E}[s_t \cos(\alpha_t)] \neq \bar{s}_t$, and as previously claimed, the estimator is biased in open space.

However, in indoor navigation, it is also necessary to take into account the effect of the geometrical constraints of the building imposed in Step 1.b.2: the net effect of Step 1.b.2 is that of selecting those particles whose parameters are admissible with respect to the geometrical constraints of the building. Hence Step 1.b.2 leads the estimator towards the correction of its bias; however, this typically causes a decrease of the effective number of particles \hat{n} , which is a working condition to be preferably avoided, if possible.

Motivated by the above considerations, the step length measurements can be scaled by a constant factor k in order to compensate for the estimator bias:

$$k = \left(1 - \frac{\sigma_\alpha^2}{2}\right)^{-1} \quad (12)$$

Since, as mentioned above, the geometrical constraints of the building have a bias correction effect, from a practical point of view, it is possible to set k to different values related to Equation (12): for instance, in Section 5, k is set to $\frac{1}{2} \left(1 - \frac{\sigma_\alpha^2}{2}\right)^{-1}$.

3.3. Revised Particle Filter

The changes presented in the previous subsections with respect to the particle filter of Subsection 2.1 provide a revised particle filtering procedure that allows one to deal with uncalibrated sensors.

The overall particle filtering algorithm can be summarized as follows:

- (1) For each particle i :
 - (1.a') Draw a sample $\hat{q}_{i,t+1}$ from the proposal distribution: draw samples $\alpha_{i,t}$ and $s_{i,t}$ from $\mathcal{N}(\alpha_t, \sigma_\alpha^2)$ and $\mathcal{N}(ks_t + \hat{s}_{b,i,t}, \sigma_s^2)$, respectively; then, compute $\hat{q}_{i,t+1}$ with Equation (3).
 - (1.b.1) If $\hat{q}_{i,t+1}$ violates the geometrical constraints of the building, then resample $\hat{q}_{i,t+1}$ as follows: draw a rotation angle sample $\hat{\alpha}_{b,t}$ from $\mathcal{U}(-\alpha_{\max}, \alpha_{\max})$, and rotate $(\hat{q}_{i,t_{s,j}}, \dots, \hat{q}_{i,t_{e,j}})$ of an angle $\hat{\alpha}_{b,t}$ around the vertical axis passing through $\hat{q}_{i,t_{e,j-1}}$.
 - (1.b.2) Set $\hat{w}_{i,t+1} = 0$ if $\hat{q}_{i,t+1}$ violates the building's geometrical constraints; otherwise, set the weight $\hat{w}_{i,t+1}$ as in Equation (4).
- (2) Scale the particle weights in order to normalize their sum to one: $\sum_{i=1, \dots, n} \hat{w}_{i,t+1} = 1$.

(3) Resample n particles from Equation (5) and set $w_{i,t+1} = 1/n$.

Notice that, similarly to the original filter of Subsection 2.1, the computational complexity of the revised particle filter is linear with respect to the number of particles n . However, its computational burden is increased with respect to the original version (by 10%–20%, approximately, in the simulations of Section 5), mostly because of Step 1.b.1.

4. Auto-Calibration of the Magnetic Sensor

The algorithm presented in the previous section allows one to obtain good tracking results also in the presence of uncalibrated sensors (as shown in the simulations of Section 5). Nevertheless, the use of calibrated sensors, if possible, is obviously preferred. This motivates the presentation in this section of an auto-calibration procedure for the magnetic sensor.

The rationale is that of collecting magnetic sensor measurements during the regular use of the device (*i.e.*, while the particle filter of Section 3 is running): compute the calibration of the magnetic sensor as soon as possible, and introduce the calibration model in the particle filter in order to improve its tracking ability.

Since the navigation problem has been considered as a 2D tracking problem and the heading direction has been assumed to be (approximately) fixed with respect to the local coordinate system, then the sensor calibration can be considered as a 2D calibration problem. To be more specific, since the heading direction and the vertical direction have been assumed to (approximately) correspond to the v_s and w_s axis, respectively, then the 2D calibration is performed on the (u_s, v_s) plane, which (approximately) corresponds to the horizontal plane.

Despite the above assumption, it is worth noticing that in practical applications, the attitude of the device is typically not fixed to a constant value for all of the time instants: in practice, the attitude of the device will be assumed to be only approximately fixed, *e.g.*, it oscillates around a mean orientation.

On the one hand, from the above considerations, it is clear that since the (u_s, v_s) plane does not exactly correspond to the horizontal plane for all of the time instants, then even an extremely precise calibration of the magnetic sensor on such a plane cannot exactly compensate for the magnetometer error in all of the time instants. On the other hand, a complete (3D) calibration of the magnetic sensor requires a full set of rotations of the device, which actually should be avoided in order to simplify the use of the navigation system to the user. Thus, the aim of this section is just the computation of a rough calibration that allows one to reduce the magnetometer error (even if typically not completely compensating for it). The usefulness of this rough calibration will be shown in the results of Section 5.

The calibration procedure considered here is similar to that in [26,36,37].

The following model for the magnetic sensor measurements is considered:

$$y_{m,t} = Dm_t + d + e_{m,t} \quad (13)$$

where $y_{m,t}$ is the magnetic sensor measurement vector (projected on the (u_s, v_s) plane), m_t is the magnetic field orientation vector (projected on the horizontal plane and normalized to one), $e_{m,t}$ is the measurement error, while D and d contain the calibration parameters.

The goal of this section is that of estimating D and d from the measurements $\{y_{m,t}\}_{t=1,\dots,T}$, by minimizing the sum of the squared norm of the errors:

$$(\hat{D}, \hat{d}, m_1, \dots, m_T) = \arg \min \sum_{t=1}^T \|y_{m,t} - Dm_t - d\|^2 \tag{14}$$

The solution of the above optimization problem will be computed by means of an iterative procedure that will be presented in the following.

First, in order to initialize the iterative optimization procedure, an initial estimation of the parameter values is required.

Since m_t is a vector with norm one, then an initial estimation of the calibration parameters can be computed similarly to [26]:

$$0 = \|m_t\|^2 - 1 \approx \|D^{-1}(y_{m,t} - d)\|^2 - 1 = y_{m,t}^\top A y_{m,t} + y_{m,t}^\top b + c \tag{15}$$

where $A = D^{-\top} D^{-1}$, $b = -2Ad$, $c = d^\top Ad - 1$.

The equation $y_{m,t}^\top A y_{m,t} + y_{m,t}^\top b + c \approx 0$ can be rewritten as a linear equation: $Y\theta \approx 0$, where $\theta = [vec(A)^\top b^\top c]^\top$ (where $vec(A)$ indicates the vectorized version of A (notice that A is a symmetric matrix; hence, only a set of independent parameters of A should be included in θ)), while Y is a matrix properly formed by the magnetic sensor measurements (at different time instants).

The equation $Y\theta \approx 0$ can be solved (in least squares sense) by means of the singular value decomposition (SVD) [38], yielding a solution vector $\hat{\theta}$ containing a scaled version \hat{A} , \hat{b} , \hat{c} of the parameters in A , b , c . The proper scale value h can be computed as in [26]: $h = (\hat{b}^\top \hat{A}^{-1} \hat{b} - \hat{c})^{-1}$. Then, with a slight abuse of notation, \hat{A} , \hat{b} , \hat{c} are redefined as follows: $\hat{A} = h\hat{A}$, $\hat{b} = h\hat{b}$, $\hat{c} = h\hat{c}$.

The initial estimates \hat{D} and \hat{d} of the calibration parameters can be obtained as follows:

$$\hat{A}^{-1} = \hat{D} \hat{D}^\top \tag{16}$$

$$\hat{d} = -\frac{1}{2} \hat{A}^{-1} \hat{b} \tag{17}$$

where the factorization of the \hat{A}^{-1} matrix (which can be computed for instance by means of the Cholesky factorization [38] or with the SVD) is not unique. Indeed, let \hat{D} be the matrix resulting from the Cholesky factorization of \hat{A}^{-1} ; then, the factorization $\hat{A}^{-1} = (\hat{D}R)(R^\top \hat{D}^\top)$ holds, as well (for each rotation matrix R).

Notice that, analogously to [36], an estimate \hat{R} of the rotation matrix R can be obtained by solving an orthogonal Procrustes problem [39]. However, such an estimate of \hat{R} will not be necessary in the following.

Once initial estimates of the involved parameters are provided as above, then the following two steps are iteratively executed (until convergence) in order to compute the solution of Equation (14):

- (1) Consider the values of \hat{D} and \hat{d} as fixed, and compute $\hat{m}_t = \hat{D}^{-1}(y_{m,t} - \hat{d})$, for $t = 1, \dots, T$. Then, redefine \hat{m}_t , for each t , by normalizing its norm to one: $\hat{m}_t = \frac{\hat{m}_t}{\|\hat{m}_t\|}$.
- (2) Consider the values of $\{\hat{m}_t\}_{t=1,\dots,T}$ as fixed and compute \hat{D} and \hat{d} by minimizing:

$$\sum_{t=1}^T \|\hat{D} \hat{m}_t - y_{m,t} + \hat{d}\|^2 \tag{18}$$

Since, in this step, both $\{\hat{m}_t\}_{t=1,\dots,T}$ and $\{y_{m,t}\}_{t=1,\dots,T}$ are considered as known, then the (least squares) solution of the above problem can be easily computed in matrix form, for instance, by means of the Moore–Penrose pseudo-inverse [38].

Notice that, as usual, when dealing with iterative optimization techniques, the convergence of the proposed two-step procedure to the global minimum of Equation (14) is not guaranteed.

The sensor model resulting from the magnetometer calibration procedure can be introduced in the particle filter presented in Section 3: when the calibration results are available, calibrated measurements are substituted to the $\{\alpha_t\}$, and Step 1.b.1 of the revised particle filter can be discarded (or can be executed, but setting α_{\max} to a smaller value).

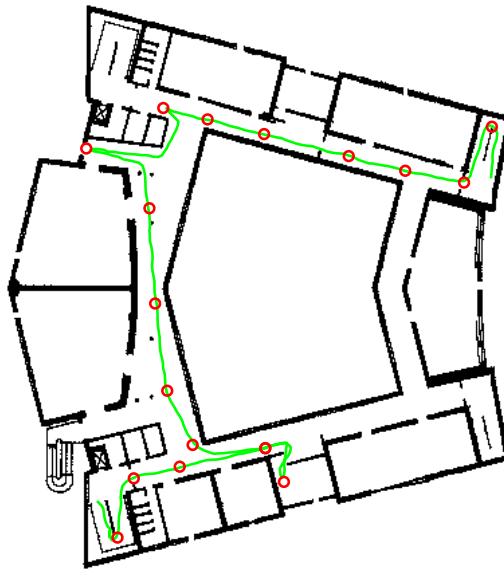
Since the use of a calibrated magnetometer can improve the performance of the particle filter, the calibration model should be used as soon as possible, *i.e.*, as soon as it provides reliable results. The results of the calibration procedure presented above can be considered (roughly) reliable, provided that a minimal number of measurements (e.g., 20 in our simulations) are in all of the octants of the (u_s, v_s) plane: once such conditions are achieved, the resulting calibration model can be used in the tracking procedure. Then, the calibration procedure shall be repeated later during the tracking of the device in order to reduce the calibration errors due to the possible small number of measurements in certain octants.

5. Experimental Validation and Discussion

The experimental validation of the method proposed in the previous sections has been assessed using a low-cost smartphone, Huawei Sonic U8650, in a building at the University of Padova. The considered smartphone is provided with a three-axis accelerometer (Bosh, BMA150, resolution 0.15 m/s^2 , maximum range $\pm 39.24 \text{ m/s}^2$), a three-axis magnetic field sensor (Asahi Kasei, AK8973, resolution $0.0625 \mu\text{T}$, maximum range $\pm 2000 \mu\text{T}$), and a Wi-Fi receiver 802.11 b/g/n. In the application used for the simulations presented in this Section, sensor measurements are acquired when changed values are detected by the system (e.g., by means of the Android sensor event method *onSensorChanged*); consequently, sensor data rates are not constant. In the simulations presented here, the approximate values of the average data rates are: 5 Hz for the accelerometer, 7 Hz for the magnetometer (since several magnetometer measurements are typically available for each human step, the heading direction is computed as the average angle obtained from such measurements) and 0.2 Hz for the Wi-Fi network (notice that such values are just indicative, because real sensor data rate values can be influenced by many factors).

A set of 30 check points has been distributed along a track involving two floors of the building: Figure 2 shows part of the trajectory (green solid line) and the check points positions (red circles) along the track. Other different tracks have been considered, as well: despite the numerical results being obviously slightly different, the performance trend is analogous to that presented here. In this sense, the case shown here can be considered as a general case (representative of the typical performance of the method).

Figure 2. Path traveled by the volunteers on the first floor of the building (green solid line) and check point positions (red circles) along the track.



In order to make the results independent of a specific stride, experimental data have been collected by three volunteers, two men and one woman, with heights from 1.65 m to 1.85 m, which completed the track with a number of steps ranging from 300 to 350. Actually, the results obtained by using the data collected by the three volunteers are quite similar: since no peculiarity has been found in the results of the tree datasets, the results presented in the following are obtained as mean values of those of the three cases.

Table 1 shows the positioning error results (root mean square error (RMSE)) obtained by using the device with uncalibrated sensors. More specifically, Table 1 compares the results obtained with the proposed revised particle filter presented in Section 3 with those obtained with the particle filter of Subsection 2.1. The particle filter of Subsection 2.1 has been run with $n = 100$ (in (a) and (d)), $n = 200$ (in (b) and (e)) and $n = 500$ (in (c) and (f)). Furthermore, two cases for the step lengths used in the particle filter have been considered: the length estimated by the sensor measurements (in (a), (b) and (c)), and the measured length corrected by the k factor (in (d), (e) and (f)). The results shown in Table 1 are the performance improvement (in percent) obtained by using the particle filter presented in Section 3 (with $n = 100$) with respect to that of Subsection 2.1. Since the results obtained by means of the particle filter are non-deterministic (*i.e.*, they depend on the specific realization of the random noise used to run the filter), the reported results have been obtained by averaging the results of 100 independent Monte Carlo simulations.

The standard deviations σ_s and σ_α have been set to proper values accordingly to our simulations: $\sigma_s = 0.15$ m and $\sigma_\alpha = \pi/6$.

Then, Table 2 shows the results obtained on the same track considered in Table 1, but using a (roughly) calibrated magnetic sensor. Calibration of the magnetic sensor has been performed as described in Section 4 by using 300 magnetic sensor sample measurements, $\{y_{m,t}\}_{t=1,\dots,300}$. The parameter values considered in Table 2 are the same considered in Table 1. Similarly to Table 1, the reported results have been obtained by averaging the results of 100 independent Monte Carlo simulations.

Table 1. Performance improvement (on the RMS positioning error) by using the revised particle filter with the uncalibrated magnetic sensor.

(a) $n = 100$	(b) $n = 200$	(c) $n = 500$	(d) $n = 100, k_{s_t}$	(e) $n = 200, k_{s_t}$	(f) $n = 500, k_{s_t}$
51.6%	55.2%	55.2%	42.7%	40.0%	55.3%

Table 2. Performance improvement (on the RMS positioning error) by using the revised particle filter with the calibrated magnetic sensor.

(a) $n = 100$	(b) $n = 200$	(c) $n = 500$	(d) $n = 100, k_{s_t}$	(e) $n = 200, k_{s_t}$	(f) $n = 500, k_{s_t}$
65.9%	62.1%	58.5%	45.5%	35.9%	33.6%

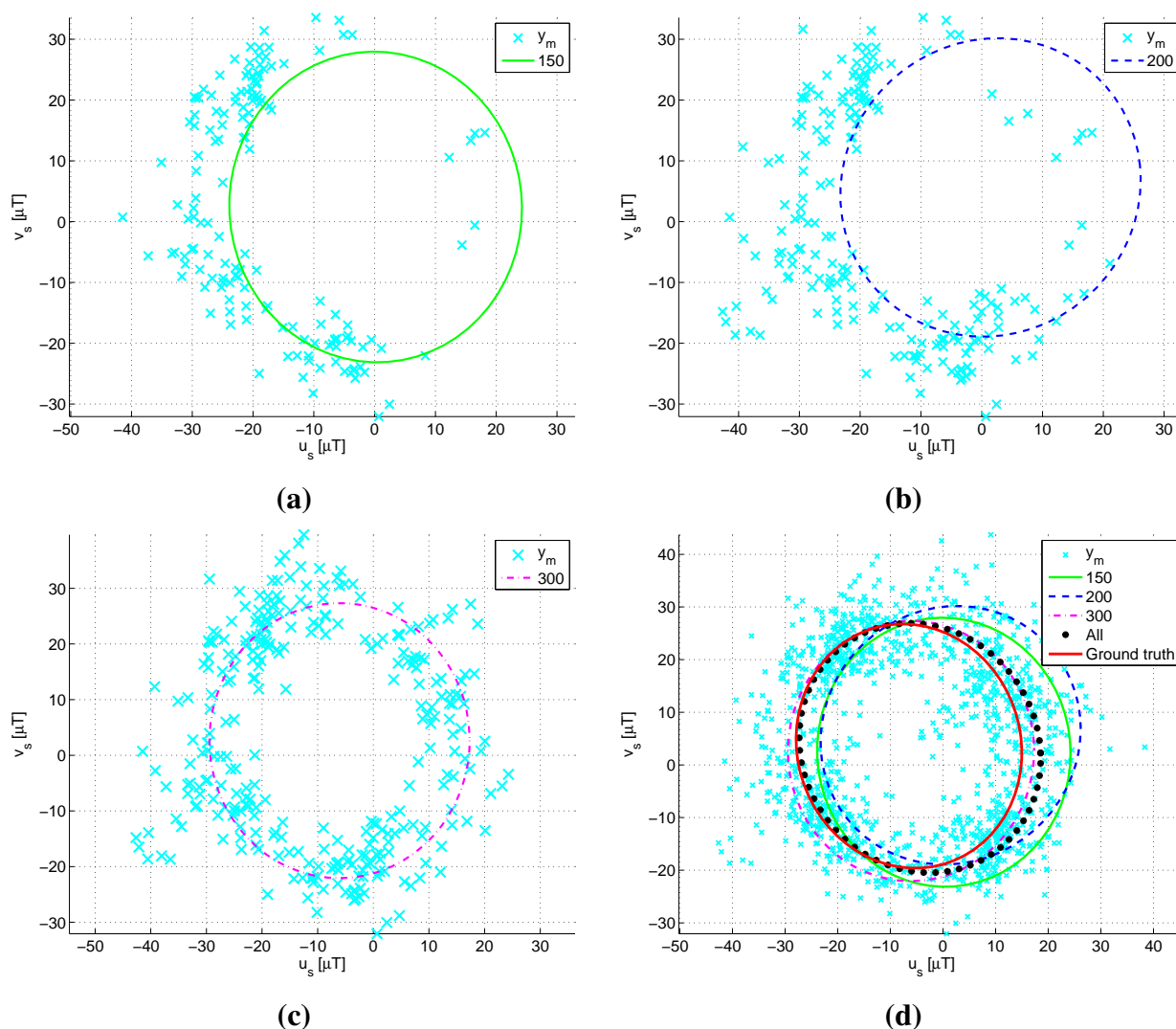
Figure 3 shows the magnetic sensor measurements and the corresponding fitting ellipse obtained with the calibration procedure described in Section 4. Calibration results are shown varying the number of sensor measurements used in the calibration process (sensor measurements are shown as cyan x-marks in Figure 3): the results obtained with 150 (top-left, solid green line), 200 (top-right, dashed blue line) and 300 (bottom-left, magenta dashed–dotted line) measurements are compared in the bottom-right sub-figure with those obtained with 1000 measurements (black stars) and with the fitting ellipse obtained with a ground truth calibration (red solid bold line). The fitting ellipse in the ground truth calibration corresponds to that of the (u_s, v_s) plane, which, during the ground truth calibration process, was coincident with the horizontal plane.

Some considerations are now in order. First, notice that the results of Table 2 have been obtained by using a roughly calibrated sensor, *i.e.*, calibrated by using 300 sensor measurements. As shown in Figure 3 (bottom-right), 300 sensor measurements do not allow one to obtain a perfect calibration. However, since, in practice, the assumption of maintaining the (u_s, v_s) plane of a handheld device in an horizontal position can be only approximately satisfied, even with more measurements, the calibration results will typically differ from the ground truth ones: indeed, as shown in the bottom-right sub-figure of Figure 3, the fitting ellipse obtained by means of 300 and 1000 measurements are quite similar, but they both differ from the ground truth fitting ellipse (red solid bold line).

Nevertheless, actually, the use of a roughly calibrated magnetic sensor can allow one to improve the performance of the tracking system: Figure 4 shows an example of unconstrained trajectories (*i.e.*, obtained by setting $\alpha_{i,t} = \alpha_t$, $s_{i,t} = s_t$ and without imposing the building physical constraints), where it is possible to compare the results obtained by using the uncalibrated magnetic sensor (blue x-marks), the magnetic sensor calibrated with 300 measurements (red x-marks, left sub-figure) and the magnetic sensor with the ground truth calibration (red x-marks, right sub-figure). As shown in the figure, the trajectory obtained with the magnetic sensor calibrated with 300 measurements is quite similar to the one obtained with the ground truth calibrated sensor. Since the trajectories shown in Figure 4 have been obtained by setting $\alpha_{i,t} = \alpha_t$, $s_{i,t} = s_t$, then they can be considered as (first) approximations of the average trajectories resulting from the particle filter (without imposing the building physical constraints) in the corresponding considered conditions: in this sense, the results shown in Figure 4 can be considered as representative of the typical outcomes. This is confirmed by the results of our

Monte Carlo simulations (Tables 1 and 2), where the use of the proposed online calibration allows one to reduce by 30%, approximately, the estimation error of both the proposed algorithm and the original particle filter.

Figure 3. Calibration of the magnetic sensor. Number of measurements used in the calibration: (a) 150, (b) 200, (c) 300. (d) Comparison of calibration results obtained with 150 (solid green line), 200 (dashed blue line), 300 (magenta dashed-dotted line), 1000 (black stars) measurements, with ground truth calibration (red solid bold line).

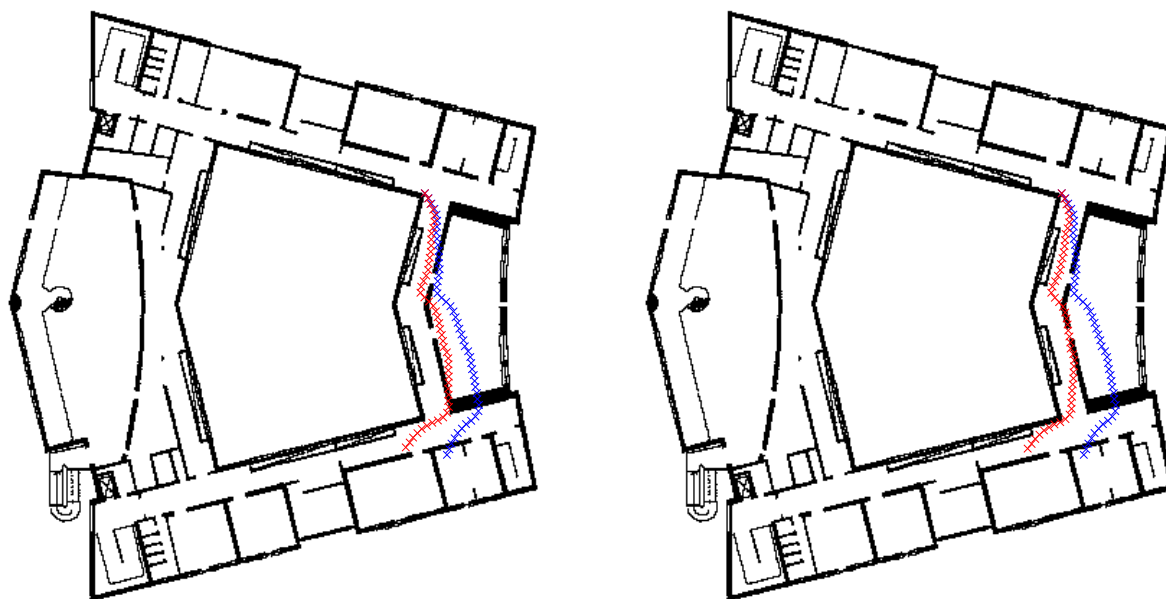


The results of Tables 1 and 2 show that the use of step lengths corrected by the factor k allows one to significantly reduce the tracking error of the particle filter. Furthermore, in all of the considered conditions, the revised particle filter proposed in Section 3 outperforms the other considered filtering approaches: despite being more clear in the uncalibrated case, its convenience is significant in the calibrated case, as well. This is probably due to the correction of local changes of the magnetic field (which yield local heading direction biases) that can be obtained by means of the proposed algorithm.

As expected, in the calibrated case, the improvement achieved by using the proposed algorithm decreases when n becomes larger; however, it is still significant even when the number of particles considered in the original particle filter (Subsection 2.1) becomes much larger than that of the revised

particle filter of Section 3. In the uncalibrated case, since measurement errors are correlated, the original particle filter (Subsection 2.1) does not significantly improve increasing the number of particles from $n = 100$ to $n = 500$.

Figure 4. Comparison of unconstrained trajectories: trajectory obtained using uncalibrated (blue x-marks) and calibrated magnetic sensor (red x-marks). Different calibration models are considered: computed by using 300 magnetic sensor measurements (**left**), and ground truth sensor model (**right**).



After the online calibration, the proposed algorithm achieves an average RMSE of (approximately) 3.5 m in the presented simulations by using $n = 100$ particles. Such performance is expected to improve by increasing the number of particles. However, considering that a larger number of n would significantly increase the computational load of the algorithm, which, according to our simulations, is approximately 10%–20% larger than in the particle filter of Subsection 2.1 with the same number of particles, $n = 100$ (however, notice that the computational load changes at each run of the algorithm, e.g., depending on the number of violated constraints).

According to the results obtained in the considered simulations, the particle filter approach proposed in Section 3 allows one to significantly improve the positioning results obtained with the filter of Subsection 2.1 in the case of uncalibrated embedded sensors and mobile device carried by the user's hand (notice that in [17,34], the use of the particle filter described in Subsection 2.1 was proposed for a different system, where sensors were integrated in the user's shoe). Such performance improvement has been obtained with a relatively small increase of the computational burden of the filter. Better results can be obtained by considering a larger number of particles n ; however, issues with the real-time implementation of the system will arise for $n = \bar{n}$, with \bar{n} sufficiently large. The actual value of \bar{n} depends on the computational capabilities of the considered device ($n \approx 100$ particles can be used for real-time applications in standard mobile phones).

Notice also that despite the proposed approach allowing the use of a mobile device carried by the user's hand, the obtained results might be improved by imposing more restrictions on the system (e.g., fixing the device to the user body, for instance to the shoe, as in [17]), or by integrating other sensors (e.g., the integration of camera measurements has been considered in [14]). Furthermore, the considered approach allows a partial, but not complete, freedom of movement to the user's hand.

Motivated by the above considerations, our future works will aim to improve the proposed method, while preserving the use of a relatively small number of particles:

- The presented approach considers two different working conditions (*i.e.*, before and after the magnetic sensor calibration results are available). However, even if the magnetometer calibration results are not yet available, the data collected can be used to improve the algorithm presented in Section 3, e.g., in Section 3, the guessed bias angles $\hat{\alpha}_{b,t}$ and $\hat{\alpha}_{b,t'}$ are chosen independently; instead, it is clear that they are (partially) related accordingly to the sensor model: the selection of the bias angle $\hat{\alpha}_{b,t'}$ should be done accordingly to the data already collected.
- An extension of the proposed approach should be considered in order to let the heading direction freely vary with respect to the local coordinate system.

Finally, the presented navigation algorithm will be integrated in a mobile mapping system in order to provide accurate 3D photogrammetric reconstructions of the environment ([40–45]) based on the use of the camera embedded in the mobile device.

6. Conclusions

In this paper, a method for smartphone-based navigation has been proposed. The considered method is based on a particle filtering tracking approach and on the fusion of information from different uncalibrated sensors.

The proposed approach has been tested, with encouraging indoor navigation results, inside a building of the university: the considered algorithm outperforms a previously proposed approach, both in uncalibrated and in (partially) calibrated working conditions.

Some future work is planned to allow further improvement of the navigation ability of the proposed strategy and reducing the assumptions on the working conditions.

Author Contributions

All authors contributed equally in this work.

Conflicts of Interest

The authors declare no conflict of interest.

References

1. Bahl, P.; Padmanabhan, V. RADAR: An in-building RF-based user location and tracking system. In Proceedings of the INFOCOM 2000 Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies, Tel Aviv, Israel, 26–30 March 2000; Volume 2, pp. 775–784.

2. Casari, P.; Castellani, A.; Cenedese, A.; Lora, C.; Rossi, M.; Schenato, L.; Zorzi, M. The “Wireless Sensor Networks for City-Wide Ambient Intelligence (WISE-WAI)” Project. *Sensors* **2009**, *9*, 4056–4082.
3. Cenedese, A.; Ortolan, G.; Bertinato, M. Low-density wireless sensor networks for localization and tracking in critical environments. *Veh. Technol. IEEE Trans.* **2010**, *59*, 2951–2962.
4. Chang, N.; Rashidzadeh, R.; Ahmadi, M. Robust indoor positioning using differential Wi-Fi access points. *IEEE Trans. Consum. Electron.* **2010**, *56*, 1860–1867.
5. Chiou, Y.; Wang, C.; Yeh, S. An adaptive location estimator using tracking algorithms for indoor WLANs. *Wirel. Netw.* **2010**, *16*, 1987–2012.
6. Mazuelas, S.; Bahillo, A.; Lorenzo, R.; Fernandez, P.; Lago, F.; Garcia, E.; Blas, J.; Abril, E. Robust Indoor Positioning Provided by Real-Time. RSSI Values in Unmodified WLAN Networks. *IEEE J. Sel. Top. Signal Process.* **2009**, *3*, 821–831.
7. Youssef, M.; Agrawala, A. The horus WLAN location determination system. In Proceedings of the 3rd International Conference on Mobile Systems, Applications, and Services, Seattle, WA, USA, 6–8 June 2005; pp. 205–218.
8. Foxlin, E. Pedestrian tracking with shoe-mounted inertial sensors. *IEEE Comput. Graph. Appl.* **2005**, *25*, 38–46.
9. Ruiz, A.; Granja, F.; Prieto Honorato, J.; Rosas, J. Accurate Pedestrian Indoor Navigation by Tightly Coupling Foot-Mounted IMU and RFID Measurements. *IEEE Trans. Instrum. Meas.* **2012**, *61*, 178–189.
10. Barbarella, M.; Gandolfi, S.; Meffe, A.; Burchi, A. A Test Field for Mobile Mapping System: Design, Set Up and First Test Results. In Proceeding of the 7th International Symposium on Mobile Mapping Technology, Cracow, Poland, 13–16 June 2011.
11. Barbarella, M.; Gandolfi, S.; Meffe, A.; Burchi, A. Improvement of an MMS trajectory, in presence of GPS outage, using virtual positions. In Proceedings of the ION GNSS 24th International Technical Meeting of the Satellite Division, Portland, OR, USA, 20–23 September 2011; pp. 1012–1018.
12. Piras, M.; Marucco, G.; Charqane, K. Statistical analysis of different low cost GPS receivers for indoor and outdoor positioning. In Proceedings of the IEEE Position Location and Navigation Symposium (PLANS), Indian Wells, CA, USA, 4–6 May 2010; pp. 838–849.
13. Piras, M.; Cina, A. Indoor positioning using low cost GPS receivers: Tests and statistical analyses. In Proceedings of the 2010 International Conference on Indoor Positioning and Indoor Navigation (IPIN 2010), Zurich, Switzerland, 15–17 September 2010.
14. Saeedi, S.; Moussa, A.; El-Sheimy, N. Context-Aware Personal Navigation Using Embedded Sensor Fusion in Smartphones. *Sensors* **2014**, *14*, 5742–5767.
15. El-Sheimy, N.; Kai-wei, C.; Noureldin, A. The Utilization of Artificial Neural Networks for Multisensor System Integration in Navigation and Positioning Instruments. *IEEE Trans. Instrum. Meas.* **2006**, *55*, 1606–1615.
16. Lukianto, C.; Sternberg, H. Stepping–Smartphone-based portable pedestrian indoor navigation. *Arch. Photogramm. Cartogr. Remote Sens.* **2011**, *22*, 311–323.

17. Widyawan; Pirkl, G.; Munaretto, D.; Fischer, C.; An, C.; Lukowicz, P.; Klepal, M.; Timm-Giel, A.; Widmer, J.; Pesch, D.; *et al.* Virtual lifeline: Multimodal sensor data fusion for robust navigation in unknown environments. *Pervasive Mob. Comput.* **2012**, *8*, 388–401.
18. Masiero, A.; Guarnieri, A.; Vettore, A.; Pirotti, F. An indoor navigation approach for low-cost devices. In Proceedings of the 4th International Conference on Indoor Positioning and Indoor Navigation (IPIN 2013), Montbéliard, France, 28–31 October 2013.
19. Gordon, N.; Salmond, D.; Smith, A. Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEE Proc. F Radar Signal Process.* **1993**, *140*, 107–113.
20. Doucet, A.; Johansen, A. *A Tutorial on Particle Filtering and Smoothing: Fifteen Years Later*; Technical Report; Department of Statistics, University of British Columbia: Vancouver/Kelowna, BC, Canada, 2008.
21. Doucet, A.; de Freitas, N.; Gordon, N. *Sequential Monte Carlo Methods in Practice*; Springer-Verlag: New York, NY, USA, 2001.
22. Zhang, Q.; Campillo, F.; Cerou, F.; Legland, F. Nonlinear system fault detection and isolation based on bootstrap particle filters. In Proceedings of the 44th IEEE Conference on Decision and Control, 2005 and 2005 European Control Conference (CDC-ECC '05), Plaza de España Seville, Spain, 12–15 December 2005; pp. 3821–3826.
23. Dellaert, F.; Fox, D.; Burgard, W.; Thrun, S. Monte Carlo localization for mobile robots. In Proceedings of the 1999 IEEE International Conference on Robotics and Automation, Detroit, MI, USA, 10–15 May 1999; Volume 2, pp. 1322–1328.
24. Andrieu, C.; Davy, M.; Doucet, A. Efficient particle filtering for jump Markov systems. Application to time-varying autoregressions. *IEEE Trans. Signal Process.* **2003**, *51*, 1762–1770.
25. Guarnieri, A.; Pirotti, F.; Vettore, A. Low-cost mems sensors and vision system for motion and position estimation of a scooter. *Sensors* **2013**, *13*, 1510–1522.
26. Hol, J. Sensor Fusion and Calibration of Inertial Sensors, Vision, Ultra-Wideband and GPS. Ph.D. Thesis, The Institute of Technology, Linköping University, Linköping, Sweden, 2011.
27. Dorveaux, E.; Vissière, D.; Martin, A.; Petit, N. Iterative calibration method for inertial and magnetic sensors. In Proceedings of the 48th IEEE Conference on Decision and Control, Shanghai, China, 15–18 December 2009; pp. 8296–8303.
28. Bonnet, S.; Bassompierre, C.; Godin, C.; Lesecq, S.; Barraud, A. Calibration methods for inertial and magnetic sensors. *Sens. Actuators A Phys.* **2009**, *156*, 302–311.
29. Schiavone, G.; Desmulliez, M.; Walton, A. Integrated Magnetic MEMS Relays: Status of the Technology. *Micromachines* **2014**, *5*, 622–653.
30. Kuo, J.T.W.; Yu, L.; Meng, E. Micromachined Thermal Flow Sensors—A Review. *Micromachines* **2012**, *3*, 550–573.
31. Hautefeuille, M.; O’Flynn, B.; Peters, F.H.; O’Mahony, C. Development of a Microelectromechanical System (MEMS)-Based Multisensor Platform for Environmental Monitoring. *Micromachines* **2011**, *2*, 410–430.
32. Au, A.K.; Lai, H.; Utela, B.R.; Folch, A. Microvalves and Micropumps for BioMEMS. *Micromachines* **2011**, *2*, 179–220.

33. Jahn, J.; Batzer, U.; Seitz, J.; Patino-Studencka, L.; Gutiérrez Boronat, J. Comparison and evaluation of acceleration based step length estimators for handheld devices. In Proceedings of the 2010 International Conference on Indoor Positioning and Indoor Navigation (IPIN), Zurich, Switzerland, 15–17 September 2010; pp. 1–6.
34. Widyawan; Klepal, M.; Beauregard, S. A Backtracking Particle Filter for Fusing Building Plans with PDR Displacement Estimates. In Proceedings of the 5th Workshop on Positioning, Navigation and Communication (WPNC 2008), Hannover, Germany, 27 March 2008; pp. 207–212.
35. Douc, R.; Cappe, O. Comparison of resampling schemes for particle filtering. In Proceedings of the 4th International Symposium on Image and Signal Processing and Analysis (ISPA 2005), Southampton, UK, 15–17 September 2005; pp. 64–69.
36. Vasconcelos, J.; Elkaim, G.; Silvestre, C.; Oliveira, P.; Cardeira, B. Geometric Approach to Strapdown Magnetometer Calibration in Sensor Frame. *IEEE Trans. Aerosp. Electron. Syst.* **2011**, *47*, 1293–1306.
37. Liu, Y.; Li, X.; Zhang, X.; Feng, Y. Novel Calibration Algorithm for a Three-Axis Strapdown Magnetometer. *Sensors* **2014**, *14*, 8485–8504.
38. Golub, G.; Loan, C.V. *Matrix Computations*; Johns Hopkins University Press: Baltimore, MD, USA, 1989.
39. Gower, J.; Dijkstra, G. *Procrustes Problems*; Oxford University Press: New York, NY, USA, 2004.
40. Hartley, R.; Zisserman, A. *Multiple View Geometry in Computer Vision*; Cambridge University Press: Cambridge, UK, 2003.
41. Ma, Y.; Soatto, S.; Kosecka, J.; Sastry, S. *An Invitation to 3-D Vision: From Images to Geometric Models*; Interdisciplinary Applied Mathematics: Imaging, Vision, and Graphics Volume 26; Springer: New York, NY, USA, 2004.
42. Agarwal, S.; Snavely, N.; Seitz, S.; Szeliski, R. Bundle Adjustment in the Large. In Proceedings of the European Conference on Computer Vision, Lecture Notes in Computer Science, Crete, Greece, 5–11 September 2010; Volume 6312, pp. 29–42.
43. Masiero, A.; Guarnieri, A.; Vettore, A.; Pirotti, F. An ISVD-based Euclidian structure from motion for smartphones. *ISPRS—Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2014**, *XL-5*, 401–406.
44. Forlani, G.; Giussani, A.; Scaioni, M.; Vassena, G. Target detection and epipolar geometry for image orientation in close-range photogrammetry. *ISPRS—Int. Arch. Photogramm. Remote Sens.* **2012**, *XXXI-5*, 518–523.
45. Feng, T.; Liu, X.; Scaioni, M.; Lin, X.; Li, R. Real-time landslide monitoring using close-range stereo image sequences analysis. In Proceedings of the 2012 International Conference on Systems and Informatics (ICSAI), Yantai, China, 19–20 May 2012; pp. 249–253.

Appendix

If the (initial) heading direction with respect to the local coordinate system is not conventionally fixed *a priori*, it can be estimated from the first steps of the user, as described in the following. First,

the system is assumed to be switched on while the device is stationary and oriented according to its ideal orientation of use with respect to the (initial) heading direction: the value of the gravitational acceleration g is measured during this phase. The first K user's steps ($K = 2$ in the simulations of Section 5) are assumed to be taken along a straight line, and the device is kept in the ideal orientation of use: then, ideally the accelerations detected by the sensor (in an inertial system) should lie on a plane, that has a simple basis formed by g and the unit vector corresponding to the heading direction. In practical operating conditions, in order to reduce the effect of measurement noise (and other non-ideal conditions), such a plane is estimated as that generated by the first two principal directions of a principal component analysis of the measured accelerations. Then, from the above considerations, the heading direction can be easily computed from the values of g and of the basis of the plane. Finally, the sign of the heading direction (*i.e.*, if the user is going forward or backward along that direction) can be easily determined by the sign of the accelerations projected along that direction.

© 2014 by the authors; licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution license (<http://creativecommons.org/licenses/by/4.0/>).