A comparison of Pedestrian Dead-Reckoning algorithms using a low-cost MEMS IMU

Conference Paper · September 2009

DOI: 10.1109/WISP.2009.5286542 · Source: IEEE Xplore

4 authors, including:

Antonio Ramón Jiménez
Spanish National Research Council
92 PUBLICATIONS **1,545** CITATIONS
SEE PROFILE

Fernando Seco
Spanish National Research Council
85 PUBLICATIONS **1,231** CITATIONS
SEE PROFILE

Jaime Andrés Guevara
Universidad del Cauca
23 PUBLICATIONS **437** CITATIONS
SEE PROFILE

A Comparison of Pedestrian Dead-Reckoning Algorithms using a Low-Cost MEMS IMU

A.R. Jiménez, F. Seco, C. Prieto and J. Guevara
Instituto de Automática Industrial. Consejo Superior de Investigaciones Científicas.
Ctra. Campo Real km. 0.2; La Poveda, 28500, Arganda del Rey.
Madrid (Spain).
Telephone: (34) 918711900, Fax: (34) 918717050
Email: arjimenez@iai.csic.es
Web: http://www.iai.csic.es/lopsi

***Abstract*—Human localization is a very valuable information for smart environments. State-of-the-art Local Positioning Systems (LPS) require a complex sensor-network infrastructure to locate with enough accuracy and coverage. Alternatively, Inertial Measuring Units (IMU) can be used to estimate the movement of a person, by detecting steps, estimating stride lengths and the directions of motion; a methodology that is called Pedestrian Dead-Reckoning (PDR). In this paper, we use low-performance Micro-Electro-Mechanical (MEMS) inertial sensors attached to the foot of a person. This sensor has triaxial orthogonal accelerometers, gyroscopes and magnetometers. We describe, implement and compare several of the most relevant algorithms for step detection, stride length, heading and position estimation. The challenge using MEMS is to provide location estimations with enough accuracy and a limited drift. Several tests were conducted outdoors and indoors, and we found that the stride length estimation errors were about 1%. The positioning errors were almost always below 5% of the total travelled distance. The main source of positioning errors are the absolute orientation estimation.**[1]

I. Introduction

Ambient intelligence aims to change the way people will interact with their environment. It pursues the idea of creating an omnipresent and imperceptible "friend" who is able to help us whenever required. Further research in artificial intelligence and sensor network technology, is needed to achieve this goal. From the sensor point-of-view Local Positioning Systems (LPS) are being investigated, using ultrasound, radio or vision technology [1], but in some cases beacon-free solutions are preferable since they do not depend on a pre-installed infrastructure.

During the last decade several beacon-free methodologies have been proposed for accurate person's position estimation based on inertial sensors [2], [3], [4], [5], [6], [7], [8], [9]. These methodologies, often called Pedestrian Dead-Reckoning (PDR) solutions, integrate step lengths and orientation estimations at each detected step, so as to compute the absolute position and orientation of a person. Some PDR approaches assume a smooth walk on horizontal surfaces, and others are valid for uneven terrain with complicated gait patterns. PDR has been proposed for a large range of applications, such as defense, emergency rescue workers, smart offices, and so on. PDR positioning accuracy, normally ranges from 0.5% to 10% of the total travelled distance, but this figures strongly depend on the algorithm implemented and the particular inertial sensor technology employed.

Inertial Measurement Units (IMU), normally contain several accelerometers, gyroscopes, magnetometers and even pressure sensors. The IMU sensors in aerospace applications, based on gimballed sensors or laser based gyroscopes, are bulky but provide a very accurate estimation with a limited drift [9]. The size and performance of an inertial sensor are linearly dependent parameters, so the smaller the sensor the lower performance is expected. Low- size and weight units such as those based on Micro-Electro-Mechanical (MEMS) sensors are becoming very popular, but they have a significant bias and therefore suffer large drifts after integration.

This paper describes, implements, and compares several of the most important algorithms for step detection (section III), stride length (section IV), heading and position estimation (section V). Several tests are conducted with a MEMS IMU attached to the foot of a person.

II. Inertial Measuring Unit

A. IMU Description

We use a commercially available IMU, model MTi from Xsens Technologies B.V (Enschede, The Netherlands). Figure 1 shows this sensor. Its size is 58x58x22 mm (WxLxH), and it weights 50 grams.

The IMU has three orthogonally-oriented accelerometers, three gyroscopes and three magnetometers. The accelerometers and gyroscopes are MEMS solid state with capacitative readout, providing linear acceleration and rate of turn, respectively. Magnetometers use a thin-film magnetoresistive principle to measure the earth magnetic field.

The performance of each individual MEMS sensor within the MTi IMU are summarized in table I. They suffer from a significant bias, and this bias also varies over time, so PDR algorithms have the challenge of avoiding excessive error accumulation (drift) during integration.

The MTi sensor has a built-in algorithm that provides the absolute heading and attitude of the unit, which is expressed as

[1]WISP 2009 SPECIAL SESSIONS: Localization in Smart Environments

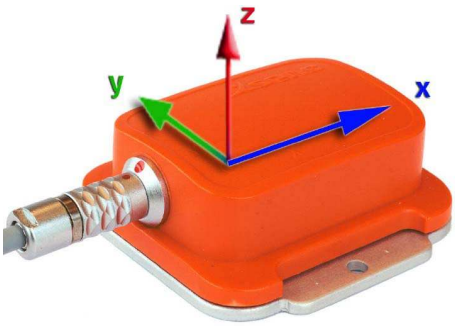

Fig. 1. MTi Xsens IMU with annotated sensor cartesian coordinates.

	accelerometers	gyroscopes	magnetometers
Axes	3	3	3
Full Scale (FS)	$\pm 50\ m/s^2$	$\pm 300\ deg/s$	± 750 mGauss
Linearity	0.2% of FS	0.1% of FS	0.2% of FS
Bias stability	$0.02\ m/s^2$	$1\ deg/s$	0.1 mGauss
Bandwidth	30 Hz	40 Hz	10 Hz
Max update rate	512 Hz	512 Hz	512 Hz

TABLE I
PERFORMANCE OF INDIVIDUAL SENSORS IN XSENS IMU

the rotation matrix $\mathbf{R}_{GS}$. It can be used to directly transform the readings from the sensor (S) to the global (G) cartesian coordinates frames. The typical absolute orientation errors are summarized in table II. Performance is quite good whenever the earth magnetic field is not disturbed, for example by metallic objects, power lines, personal computers, or any device containing electro-magnetic motors.

B. IMU placement

Several IMU locations have already been tested, e.g. the waist, trunk, leg, foot or even the head [6]. The waist or trunk locations are probably the less intrusive IMU placements, and also the most reliable position for heading estimation using gyroscopes or magnetometers [6]. However, the foot mount has decisive advantages: 1) It is applicable the zero velocity update (ZUPT) strategy to diminish drifts after integrating accelerations [2][3], and 2) the step detection is robustified.

In this paper we use the IMU mounted on the foot. Figure 2 shows the Xsens sensor fixed, using the shoe laces, to the right foot of a person. The exact position and orientation of the IMU on the foot is not important, because many algorithms only work with the magnitude of sensor readings, and if necessary for a different processing, the individual sensor readings could be transformed to the world coordinates system.

Static accuracy (roll/pitch)	<0.5 deg
Static accuracy (heading)[1]	<1 deg
Dynamic accuracy	2 deg RMS
Angular resolution	0.05 deg

[1] in homogeneous magnetic environment

TABLE II
PERFORMANCE OF ATTITUDE AND HEADING AS PROVIDED BY XSENS FUSION ALGORITHM IN MATRIX $\mathbf{R}_{GS}$.

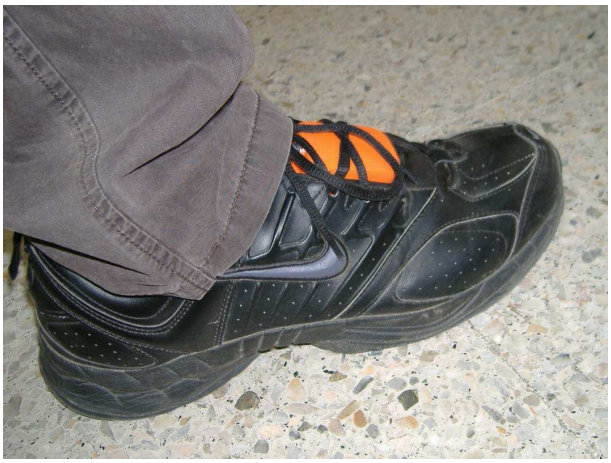

Fig. 2. Xsens IMU attached to the right foot using the shoe's laces.

III. STEP DETECTION

A. Preliminary walking tests

Three different walking tests were done to evaluate the performance of some algorithms for step detection:

1) A one-way straight walk (120 meters long)
2) A go and return walk (60 m long with a 180º turn)
3) A rectangular path (30 x 5 m; 70 meters in total)

Each of them were done at three different walking speed (slow, normal, and fast), and repeated three times to check the repetitivity of results. These initial tests were conducted over an even terrain that was far from metallic objects or motors in order to avoid magnetic perturbations. As a reference for subsequent performance evaluation, we register the number of steps needed to finish the predefined trajectory, and the actual travelled distance.

B. Captured Signals

The IMU was configured to capture sensor samples at 100 Hz. Figure 3 shows the raw sensor data captured in a walking test, 30 meters long with a 180º turn, that required 37 steps. We clearly see the step occurrences in any of the three types of sensors (acceleromets, gyroscopes and magnetometers). The 180º turn is easily visualized in the middle of the signals. Note that the signal magnitudes are very low at the start and the end of the motion; this makes difficult the detection of all steps using simple threshold-based algorithms.

C. Step Detection using accelerations

Algorithms in the literature for step detection rely on basic data processing techniques (filtering, magnitudes, local variances and thresholding). Most of them use the data provided by accelerometers [5] or gyroscopes [2]. We implemented state of the art step detection algorithms with some slight modifications, and additionally we tested the use of magnetometers to detect steps.

The batch-mode algorithm implemented for step detection consists of the following 4 steps:

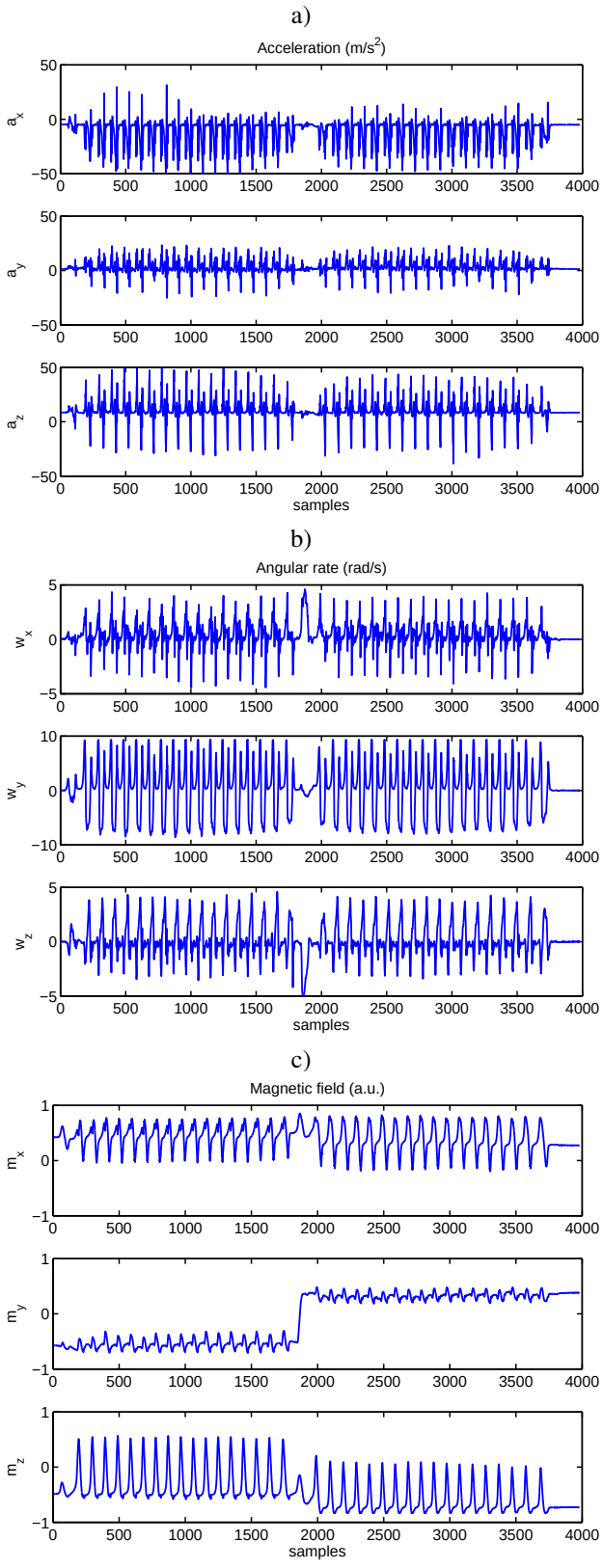

Fig. 3. Raw sensor readings for a test of 30 meters walking in one direction and returning back after a 180 degree turn: a) Acceleration, b) Gyroscope, and c) Magnetometer readings.

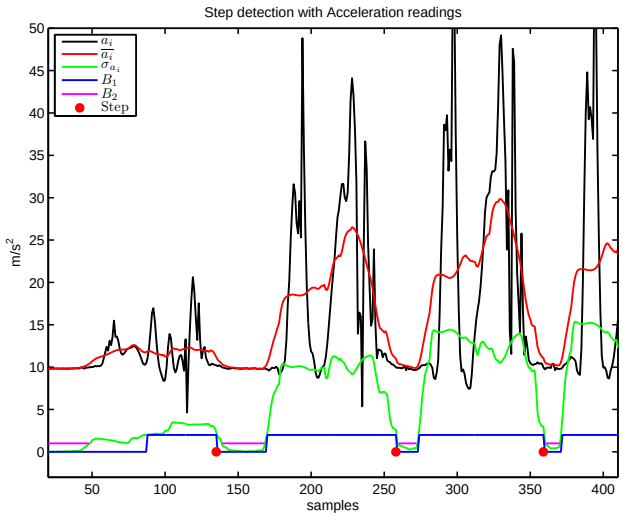

Fig. 4. Steps detection using accelerations. The detected steps are marked with red circles. Additional line plots represent intermediate processing values as explained in equations 1 to 3.

1) Compute the magnitude of the acceleration, a_i, for every sample i:

$$a_i = \sqrt{a_{x_i}^2 + a_{y_i}^2 + a_{z_i}^2}. \quad (1)$$

2) Compute the local acceleration variance, to highlight the foot activity and to remove gravity:

$$\sigma_{a_i}^2 = \frac{1}{2w+1} \sum_{j=i-w}^{i+w} (a_j - \overline{a_j})^2, \quad (2)$$

where $\overline{a_j}$ is a local mean acceleration value, computed by this expression: $\overline{a_j} = \frac{1}{2w+1} \sum_{q=i-w}^{i+w} a_q$, and w defines the size of the averaging window (w=15 samples).

3) Thresholding. A first threshold is applied to detect the swing phase with high accelerations (T1=2 m/s^2).

$$B_{1i} = \begin{cases} \text{T1} & \sigma_{a_i} > \text{T1} \\ 0 & \text{otherwise} \end{cases}. \quad (3)$$

A second threshold (T2=1 m/s^2) is used to detect the stance phase ($B_{2i} = \text{T2}$, if $\sigma_{a_i} < \text{T2}$).

4) A step is detected in sample i when a swing phase ends and stance phase starts. These two conditions must be satisfied: 1) a transition from high to low acceleration ($B_{1i-1} < B_{1i}$), and 2) there must be at least one low acceleration detection in a window of size w ahead of current sample i, i.e.: $\max(B_{2i:i+w}) = \text{T2}$.

Figure 4 shows details of this step detection process.

D. Step detection using gyroscopes and magnetometers

We implemented a step detection algorithm based on angular rate readings [2]. It initially computes the total angular rate magnitude using the three individual gyroscopic sensors, then it performs a threshold at 1 rad/s, after that it applies a median filter to remove outliers, and finally it detects transitions to a motionless state.

Sensor & Algorithm type	Real number of steps	Number of steps failed	Percentage of errors
Acceleration	955	1	0.1%
Gyroscopes	955	2	0.2%
Magnetometric	955	9	0.94%

TABLE III
PERFORMANCE OF THREE STEP DETECTION ALGORITHMS

We also defined an algorithm for step detection using the magnetometer. It starts with a high pass filter for removing the DC components of the magnetic field. After that, the remaining processing tasks are the same as in the angular rate algorithm described in last paragraph.

E. Results

The three implemented step detection algorithms were tested. The total number of steps in the whole set of tests were 955. As Table III shows, the most reliable algorithm is the one based on acceleration readings (0.1% error). The gyroscopic-based method was also quite accurate (0.2%), and even the magnetometric method (0.94%) did not perform too bad.

In general, the step detection is very reliable at continuous walk, but we found that it is more difficult to detect steps robustly at the beginning/end of motion, and at very low speeds (e.g. museum-like style of walking). For its better performance, we will use along the rest of the paper the acceleration-based method for step detection.

IV. STRIDE LENGTH ESTIMATION

It is neccessary to estimate the Stride Length (SL) at every detected step in order to calculate the total forward movement of a person while walking. SL can not be assumed to be constant, it varies significantly depending on the person, its leg length, and its walking speed and frequency. There are three main procedures to estimate a SL:

1) Foot-to-foot range measuring devices [8] (Not implementable just with IMUs).
2) Modelling human gait with inertial measurements [7].
3) Foot inertial integration (INS) with zero velocity updates (ZUPT) at stance detections [2], [3].

We implemented two SL algorithms. One, of type 2, and the other of type 3. Next sections describe them.

A. The Weiberg SL Algorithm

The algorithm proposed by Weinberg [7] assumes that SL is proportional to the bounce, or vertical movement, of the human hip. This hip bounce is estimated from the largest acceleration differences at each step. This is the Weiberg algorithm:

1) Compute the magnitude of accelerations, a_i, as in eq. 1.
2) Low-Pass filter this signal ($\tilde{a}_i = LP(a_i)$). We use a filter of order 4 and cut-off frequency at 3 Hz.
3) Estimate the SL using the Weiberg expression:

$$\text{SL}_{\text{Weiberg}_k} = K \cdot \{\max_{j=[i_{(k)} \pm w]}^{\tilde{a}_j} - \min_{j=[i_{(k)} \pm w]}^{\tilde{a}_j}\}^{1/4}, \quad (4)$$

where the maximum and minimum operations are applied over the filtered accelerations $\tilde{a}_j$ in a window of size $2w+1$ around the sample $i_{(k)}$ corresponding to the k stance detection. K is a constant that has to be selected experimentally or calibrated.

B. The ZUPT SL Algorithm

The ZUPT algorithm is considered as the most reliable and versatile method regardless of the user and displacement patterns (walk, run, side walk, cris-cross, climb) [2]. This inertial mechanization method performs zero velocity updates every time a step is detected. At foot stance the velocity is known to be zero, so the idea is to correct the linear velocities obtained after integrating the accelereometer. This is a good way to strongly attenuate the bias of accelerometers and the subsequent drift in velocity and position. This is the algorithm:

1) Transform the captured accelerations, a_i refered to the sensor frame (S), into the global north-east frame (G), using the rotation matrix: $a_i^G = \mathbf{R}_{GS} \cdot a_i$
2) Integrate these accelerations, a_i^G, to obtain the linear velocities v_i^G. Thats is: $v_i^G = v_{i-1}^G + a_i^G / f_s$, where f_s is the sampling frequency (100 Hz).
3) Correct the linear velocity v_i^G from drift by using the ZUPT update at every stance event:
   a) Compute the mean velocity value, μ, around the stance event, k, as follows: $\mu_k = \sum_{j=i_{(k)}-w}^{s_{(k)}+w} v_j^G/(2w+1)$, where $i_{(k)}$ represents the sample index of the k stance occurrence. Note that μ_k represents the velocity error accumulated at step k (it should be zero).
   b) Correct all the velocity samples in the whole step. We use a weighted linear interpolation between two consecutive stance events: $\check{v}_i^G = v_i^G - [\mu_k(i - i_{(k-1)}) + \mu_{k-1}(i_{(k)} - i)]/m_k$, where m_k is the number of samples in step k.
4) We obtain the position increment at step k, $\Delta P_k = (\Delta P_k(north), \Delta P_k(east), \Delta P_k(up))$, by integrating the corrected velocities samples $\check{v}_i^G$ in that single step:

$$\Delta P_k = \sum_{j=i_{(k-1)}}^{i_{(k)}} \check{v}_j^G / f_s. \quad (5)$$

5) The 2D SL is computed by taking the horizontal cartesian distance of the position increment:

$$\text{SL}_{\text{ZUPT}_k} = \sqrt{\Delta P_k(north)^2 + \Delta P_k(east)^2}. \quad (6)$$

C. Results

We computed the SL at every detected step in walking tests at three speeds: slow, normal and fast. The actual stride length at every step is unknown, however we know the average SL for each test (the actual travelled distance divided by the total number of steps). Figure 5 shows the SL estimations obtained with the Weiberg and ZPUT methods. We see how these estimations are not too biased from the actual SL averages.

Note that the Weinberg methodology, using a fixed K value, is valid for accurately estimating SL even at different walking

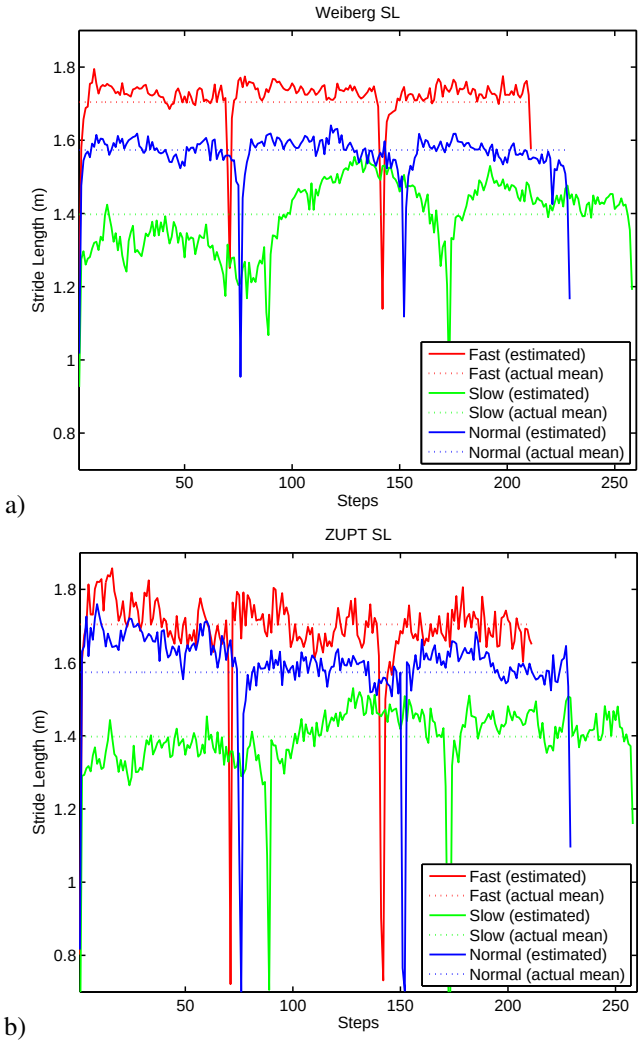

Fig. 5. Stride length estimations for straight trajectories (120 m of length) at three different walking speeds, and repeated 3 times each of them. a) Weinberg SL algorithm ($K = 0.364$) [7], b) ZUPT SL algorithm [2].

speeds. The ZUPT method gives also good results with a larger variation between consecutive SL; this is because ZUPT method does not filter any signal as Weiberg SL method does.

The total travelled distance, at a given walking speed, was estimated by integrating all the individual SL estimations. Comparing them with the actual distances we found very accurate results as summarized in table IV. Both SL methods give errors about 1% of the total travelled distance.

SL algorithm	Walking speed	Error of the total distance calculated by summing all SL	Error in percentage of the total travelled distance (360 m)
Weiberg	Slow	-1.10 m	0.30%
	Normal	-2.64 m	0.73%
	Fast	2.81 m	0.78%
ZUPT	Slow	-2.23 m	0.62%
	Normal	4.15 m	1.15%
	Fast	-3.47 m	0.97%

TABLE IV
PERFORMANCE OF TWO STRIDE LENGTH (SL) ESTIMATION ALGORITHMS (WEINBERG AND ZUPT) FOR THREE DIFFERENT WALKING SPEEDS.

V. 2D POSITION ESTIMATION

A. Positioning algorithms

We implemented three positioning algorithms. Two of them are based on the accumulation of foot displacements (estimated stride lengths SL_k) along the horizontal orientation of the foot (θ_{stance_k}) at each detected step stance . This is the algorithm:

1) Compute the IMU's X axis in global coordinates, $X^G_{\text{IMU}_{i_{(k)}}} = \mathbf{R}_{GS_{i_{(k)}}} \cdot [1,0,0]$, for every detected step k at stance occurrence $i_{(k)}$.
2) Assuming that IMU X axis is aligned with the foot heading, then the foot orientation at stance is:

$$\theta_{\text{stance}_k} = arctan\left(\frac{-X^G_{\text{IMU}_{i_{(k)}}}(\text{west})}{X^G_{\text{IMU}_{i_{(k)}}}(\text{north})}\right). \quad (7)$$

3) The horizontal position $P_k = (P_k(\text{north}), P_k(\text{east}))$ of the foot at step k is computed as

$$\begin{cases} P_k(\text{north}) = P_{k-1}(\text{north}) + \text{SL}_k \cdot cos(\theta_{\text{stance}_k}) \\ P_k(\text{east}) = P_{k-1}(\text{east}) + \text{SL}_k \cdot sin(\theta_{\text{stance}_k}), \end{cases} \quad (8)$$

where SL_k are the stride lengths estimated by any of the two method presented in last section (eqs. 4 or 6).

The third INS-ZUPT-based algorithm for foot position estimation just accumulates the positioning increments, ΔP_k, obtained with equation 5:

$$\begin{cases} P_k(\text{north}) = P_{k-1}(\text{north}) + \Delta P_k(\text{north}) \\ P_k(\text{east}) = P_{k-1}(\text{east}) + \Delta P_k(\text{east}). \end{cases} \quad (9)$$

B. Outdoors and indoors positioning results

We tested the three above-presented positioning algorithms with several trajectories both outdoors and indoors. Trajectories have a length between 100 and 320 meters, and were closed, i.e. the finish point coincides with the start position. The same paths were followed in clock-wise (CW) direction and also counter-clock-wise (CCW).

After a study of more than 30 test trajectories, the error between the final position estimation and the starting point, was in most of the cases between 5 to 15 meters, which accounts for a percentage of the total travelled distance below 5%. These final-position errors were similar for all three algorithms, but slightly worse for the INS-ZUPT method.

Figure 6 shows the estimated trajectories of an outdoor test, and figure 7 an indoor test in the main building of IAI-CSIC. We note in both figures that there is a significant rotation error in the estimated trajectories; CW for the INS-ZUPT algorithm based on eq. 9; and a slight angular error CCW in the algorithms based on SL and foot orientation at stance (eq. 8). We know that any orientation error in the latter methods can come from the misalignment between the foot and the sensor

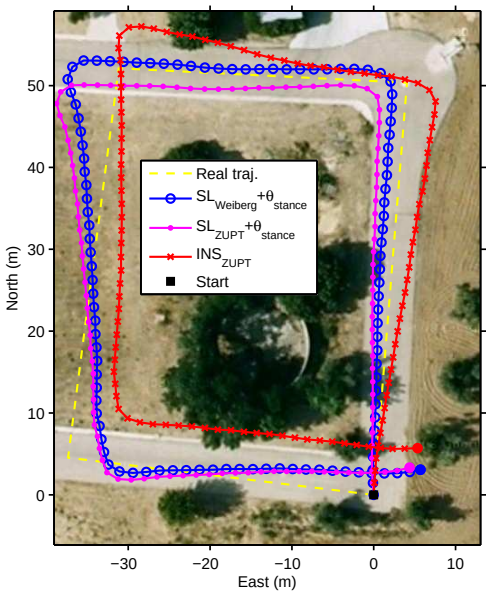

Fig. 6. An outdoors 160 meter-long closed-CCW trajectory and three position estimations using different PDR algorithms.

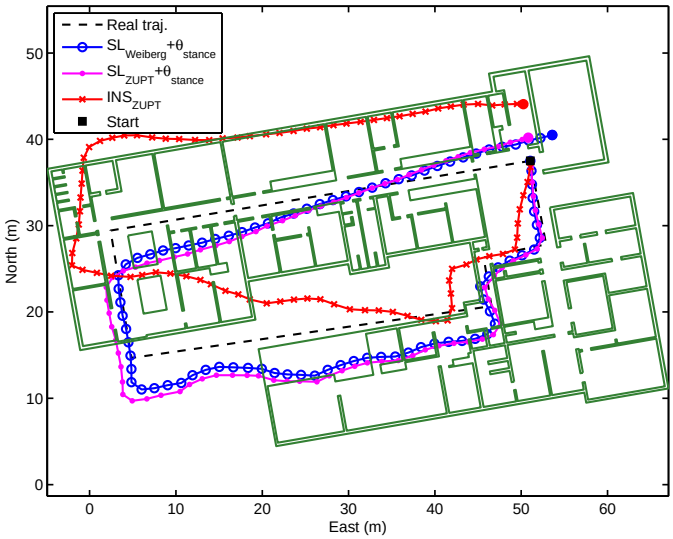

Fig. 7. A trajectory along the main building of IAI-CSIC and three position estimations using different PDR algorithms.

X axis. This unknown misalignment term could be calibrated, but we presented the raw trajectories.

Considering the shape of the trajectories, its quality is slightly worse with the INS-ZUPT method. We know that the positioning results strongly depends on the quality of orientation estimations. The INS-ZUPT method uses all orientation samples in a step (about 80), however the SL $+\ \theta$ methods use the orientation at one step sample (the sample at stance detection). We observed that orientation during swing phase are not as reliable as orientation at stance, this could explain why the SL $+\ \theta$ methods give trajectories with slightly better shape and accuracy than INS-ZUPT.

VI. Conclusion

We have described, implemented and compared some of the most relevant algorithms in the state of the art for pedestrian dead reckoning (PDR). We restricted the study to low-cost, low-performance, low-weight MEMS-based IMU sensors placed at the foot of a person. The presented results shown that MEMS IMU with the appropriated algorithms can provide good solutions for estimating human trajectories, with a drift that is proportional to the travelled distance but not to the time elapsed. The positioning errors are below 5% of the travelled distance both outdoors and indoors (although it depends on magnetic disturbances and the walking pattern).

The INS-ZUPT positioning method is normally considered as the most powerful PDR method, since it is able to satisfactorily estimate while running, with lateral walk or on uneven terrain. However we have seen that in smooth surfaces other positioning methods, based on SL plus foot orientations at stance, can be superior and more computationally efficient.

Any IMU PDR estimation method needs a periodic calibration to obtain fine positioning. This update must be done with absolute position estimates, that could be provided by LPS sensors such as those based on RFID, WiFi, UWB or ultrasound. Future work, will be focused to: 1) integrating PDR methods with LPS solutions, and 2) matching PDR estimations with local maps for absolute orientation correction. Any of these approaches should provide accurate indoor location of persons, valid in the analysis of motion in smart environments and scenarios such as home, offices, healthcare, emergencies, sports and so on.

Acknowledgment

The authors would like to thank the financial support provided by projects RESELAI (TIN2006-14896-C02-02) and LOCA (CSIC-PIE Ref.200450E430).

References

[1] A.R. Jiménez, F. Seco, C. Prieto, and J. Roa, "Tecnologías sensoriales de localización para entornos inteligentes," in *I Congreso español de informática - Simposio de Computación Ubicua e Inteligencia Ambiental, UCAmI2005 (Granada)*, 2005, pp.75-86.

[2] R. Feliz, E. Zalama and J. Gómez, "Pedestrian tracking using inertial sensors," *Journal of Physical Agents*, vol. 3 (1), pp. 35-42, 2009.

[3] B. Beauregard, "Omnidirectional Pedestrian Navigation for First Responders," in *4th Workshop on Positioning, Navigation and communication, WPNC'07, Hannover*, 2007, pp. 33-36.

[4] J. Won Kim, H.J. Jang, D. Hwang and C. Park, "A step, stride and heading determination for the pedestrian navigation system," *Journal of Global Positioning Sustems*, vol. 3 (1-2), pp.273-279, 2004.

[5] R. G. Stirling, "Development of a pedestrian navigation system using shoe mounted sensors", *Master of Science*, University of Alberta, 2003.

[6] R Stirling, J. Collin, K. Fyfe and G. Lachapelle, "An Innovative Shoe-mounted Pedestrian Navigation System", In *GNSS 2003, Graz, Austria, 22-25 April*, 2003, pp. 1-15.

[7] H. Weinberg, "Using the ADXL202 in Pedometer and Personal NavigationApplications," *Analog Devices AN-602 application Note*, 2002

[8] T.J. Brand and R.E. Phillips, "Foot-toFoot Range Measurement as an Aid to Personal Navigation,´´ in *ION 59th Annual Meeting, 23-25 June, Alburquerque, NM*, 2003, pp. 113-121.

[9] L. Ojeda and J. Borestein, "Non-GPS Navigation for Emergency Responders," In *International joint Topical Meeting on Sharing Solutions for Emergencies and Hazardous Environments, February 12-15, Salt Lake City, Utath, USA*, 2006, pp. 1-8.