

Article

Height Compensation Using Ground Inclination Estimation in Inertial Sensor-Based Pedestrian Navigation

Sang Kyeong Park and Young Soo Suh *

Department of Electrical Engineering, University of Ulsan, Namgu, Ulsan 680-749, Korea;

E-Mail: damiro76@hotmail.com

* Author to whom correspondence should be addressed; E-Mail: yssuh@ulsan.ac.kr;

Tel.: +82-52-259-1524; Fax: +82-52-259-1686.

Received: 8 July 2011; in revised form: 11 August 2011 / Accepted: 12 August 2011 /

Published: 15 August 2011

Abstract: In an inertial sensor-based pedestrian navigation system, the position is estimated by double integrating external acceleration. A new algorithm is proposed to reduce z axis position (height) error. When a foot is on the ground, a foot angle is estimated using accelerometer output. Using a foot angle, the inclination angle of a road is estimated. Using this road inclination angle, height difference of one walking step is estimated and this estimation is used to reduce height error. Through walking experiments on roads with different inclination angles, the usefulness of the proposed algorithm is verified.

Keywords: inertial sensors; angle measurement; pedestrian navigation; Kalman filter

1. Introduction

A pedestrian navigation system provides a person's location indoors or outdoors. Many different technologies are used for the pedestrian navigation such as vision [1], wireless technology [2], ultrasonic sensors [3,4], and inertial sensors [5–10]. Among them, an inertial sensor-based pedestrian navigation system computes the location using inertial sensors installed on a shoe [5–10]. This inertial navigation system can be used for first respondents and soldiers [11].

The inertial-based navigation algorithms can be classified into two groups. One is that a step length is estimated using some parameters such as one step walking time or accelerometer maximum values [5,6]. The other is that the position is computed using the inertial navigation algorithm [12], where results in [7–10] belong to this group. Basic inertial pedestrian navigation algorithms are proposed in [7,8].

In [9], inertial navigation algorithm is combined with a map matching algorithm. In [10], an HMM filter is used to estimate gait phases and detect zero velocity intervals. Also we note that the inertial navigation algorithm is also used to analyze gait for medical purposes [13,14].

This paper is related to the latter, where the inertial navigation algorithm is used. A main advantage of an inertial sensor-based algorithm is that no landmarks or devices need to be installed on the environment. On the other hand, an inertial navigation algorithm has an inherent problem that the position error grows as time elapses.

To reduce the error growth, all inertial-based algorithms employ the zero velocity updating [15]. During walking, a foot touches the ground almost periodically for a short time interval and we know the velocity of a foot during the interval is zero. This interval is called a zero velocity interval. The errors of the inertial navigation algorithm are reduced using this zero velocity interval.

However, even with the zero velocity, the error growth can be significant. Experiment results reveal that z axis position (height) error growth could be in particular significant (see left plots in Figure 8 and Figure 9). One explanation is that acceleration in the z axis could be very large when a foot touches the ground and z axis acceleration data could be saturated (see Figure 4). This will degrade the position accuracy of all three axes and the effect is most dominant in the z axis position and velocity since the acceleration is the largest in the z axis direction.

In this paper, we propose a new algorithm, which reduces the z -axis position error. We use the fact that a foot angle is almost the same as the inclination angle of a ground when a foot is on the ground (see Figure 2). If we measure a foot angle during the zero velocity (that is, when a foot is on the ground), the inclination angle of a ground can be estimated. Using the estimated inclination angle, the z -axis position error can be compensated.

The paper is organized as follows. In Section 2, basic equations of inertial sensor-based pedestrian navigation algorithm is introduced and detailed equations are given in Appendix. In Section 3, a height compensation algorithm using a foot angle estimation is proposed. In Section 4, the proposed algorithm is verified through experiments. Conclusion is given in Section 5.

2. Inertial Sensor-Based Pedestrian Navigation System

In this section, a brief introduction to an inertial navigation algorithm is given. Details about inertial navigation algorithms can be found in [12,16,17]. The inertial navigation algorithm used in this paper is from [10].

Let $r \in R^3$, $v \in R^3$ and $q \in R^4$ be a position, a velocity and attitude quaternion in the navigation coordinate frame. The navigation coordinate frame has axes with the direction north (x axis), west (y axis), and the local vertical (up, z axis). The body coordinate frame is fixed on the sensor unit. The rotation matrix associated with the quaternion q is expressed as $C(q)$. In the inertial navigation algorithm, r , v and q are estimated from inertial sensors (accelerometers and gyroscopes) and magnetic sensors.

Let ω_b , a_b , \tilde{g} and \tilde{m} be defined as follows:

- $\omega_b \in R^3$: body angular rates
- $a_b \in R^3$: body acceleration without gravitational acceleration

- $\tilde{g} \in R^3$: gravitational acceleration vector in the navigation coordinate frame
- $\tilde{m} \in R^3$: earth magnetic field vector in the navigation coordinate frame.

The sensor output equations are given by

$$\begin{aligned} y_g &= \omega_b + b_g + v_g \\ y_a &= C(q)\tilde{g} + a_b + b_a + v_a \\ y_m &= C(q)\tilde{m} + v_m \end{aligned} \quad (1)$$

where $y_g \in R^3$ is a gyroscope output, $y_a \in R^3$ is an accelerometer output, $y_m \in R^3$ is a magnetic sensor output, $b_g \in R^3$ is gyroscope bias, $b_a \in R^3$ is accelerometer bias, $v_g \in R^3$ is gyroscope sensor noise, $v_a \in R^3$ is accelerometer sensor noise, and $v_m \in R^3$ is magnetic sensor noise.

An indirect Kalman filter is used to estimate q , r and v . In an indirect filter, q , r and v are not directly estimated. Instead q , v and r are first estimated by appropriately integrating or double integrating y_g and y_a and their errors are estimated using a Kalman filter [10]. The indirect Kalman filter equations are given in Appendix.

The following assumptions are made in the paper:

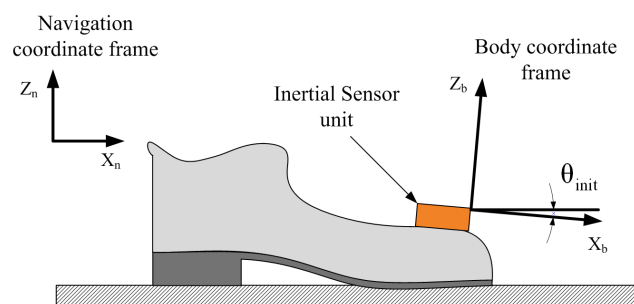
- Walking direction is only forward and no side walking and backward walking are allowed.
- There is no staircase on the walking path

The assumptions are satisfied during normal walking situations: that is, a person walks forward only and stopping is allowed.

3. Height Compensation Algorithm

The inertial sensor unit is installed on top of a shoe as in Figure 1. When a person is standing on a flat ground, the unit is not completely level and the roll angle at that time is denoted by θ_{init} .

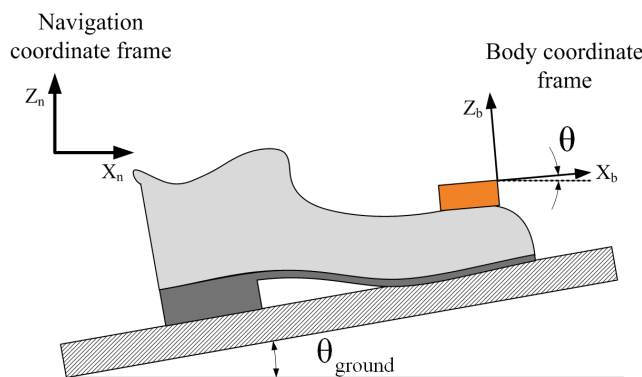
Figure 1. Initial angle θ_{init} when a foot is on a flat ground.



When a foot is on a slope with the inclination angle θ_{ground} (see Figure 2), the roll angle θ of the sensor unit is different from θ_{init} , which is the roll angle when a foot is on a flat ground. The relationship between θ and θ_{ground} is given by

$$\theta_{ground} = \theta - \theta_{init}. \quad (2)$$

Since θ_{init} is constant, we can estimate the ground inclination angle θ_{ground} once we know the roll angle θ .

Figure 2. Roll angle θ of the sensor unit when a foot is on a slope.

The roll angle θ can be computed from the attitude quaternion \hat{q} . Or θ can be also estimated using accelerometer outputs during the zero velocity interval since there is no external acceleration. From Equation (1), we obtain the following ignoring sensor noises and external acceleration

$$y_a \approx C(q) \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} = \begin{bmatrix} -\sin \theta \\ \cos \theta \sin \phi \\ \cos \theta \cos \phi \end{bmatrix} g$$

where ϕ is the pitch angle. Thus θ and ϕ can be estimated using the following:

$$\begin{aligned} \phi &= \text{atan2}(y_{a,y}, y_{a,z}) \\ \theta &= \text{atan2}(-y_{a,x}, \sqrt{y_{a,y}^2 + y_{a,z}^2}). \end{aligned} \quad (3)$$

In this paper, roll angle θ is estimated using Equation (3) during each zero velocity interval: θ is computed for each discrete time during a zero velocity interval and the averaged value is used as $\hat{\theta}$, which is an estimated value of θ . Thus $\hat{\theta}$ is updated whenever zero velocity intervals are encountered. In the update, a low pass filter is used to suppress a sudden change of $\hat{\theta}$.

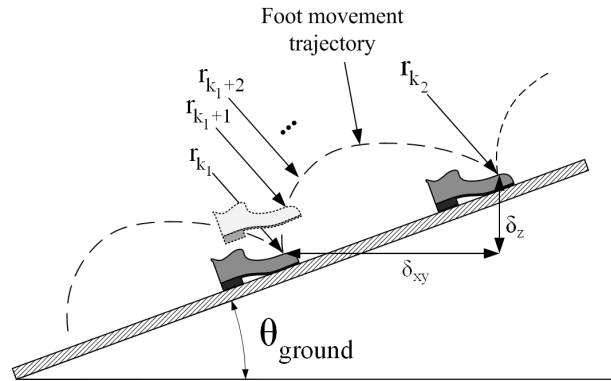
There are many methods to detect zero velocity intervals [15]. In this paper, we used both accelerometer values and force sensors (Tekscan FlexiForce Sensors), which are installed inside a shoe. Note that the measured force increases when a foot is on the ground since the human weight is applied on the sensors. We assume a discrete time i belongs to a zero velocity interval if the measured force sensor is larger than the prespecified value and θ_i (θ value computed using Equation (3) at the discrete time i) satisfies the following

$$|\theta_i - \theta_{i-1}| < 2^\circ. \quad (4)$$

Note that Equation (4) is equivalent to the condition that changes of accelerometers are small.

A typical foot movement trajectory when a person is walking on a slope is given in Figure 3. Note that r_{k_1} is the foot position at the discrete time k_1 . In this example, the foot is on the slope at the discrete time k_1 and k_2 . A person walked one step between the time k_1 and k_2 . Thus r_{k_1} is the position before one step walking and r_{k_2} is the position after the step. Let δ_{z,k_1,k_2} and δ_{xy,k_1,k_2} be the horizontal and vertical distances between r_{k_1} and r_{k_2} , respectively:

$$\begin{aligned} \delta_{xy} &= \left\| \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} (r_{k_2} - r_{k_1}) \right\|_2 \\ \delta_z &= \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} (r_{k_2} - r_{k_1}). \end{aligned} \quad (5)$$

Figure 3. Foot movement between two zero velocity intervals.

Note that δ_{z,k_1,k_2} and δ_{xy,k_1,k_2} are horizontal and vertical distances of one walking step. We will drop k_1 and k_2 subscripts in δ for simplicity.

Assuming that walking is mostly up or down along the slope, δ_z and δ_{xy} have the following relationship:

$$\delta_z \approx \tan(\theta_{ground})\delta_{xy}. \quad (6)$$

Equation (6) is used in the measurement update of the Kalman filter in Appendix. Let $\hat{\theta}_{k_1}$ be estimated θ value at time k_1 using Equation (3) and \hat{r}_{k_1} be the position estimate of the inertial navigation algorithm. Let $\hat{r}_{k_2}^-$ be the position estimate of the inertial navigation algorithm before the measurement update (that is, the zero velocity updating). Thus $\hat{r}_{k_2}^-$ is obtained by double integrating acceleration starting from time k_1 with the initial value \hat{r}_{k_1} .

From Equation (6), let $\hat{\delta}_z$ (estimate of δ_z) be defined by

$$\hat{\delta}_z = \tan(\hat{\theta}_{k_1} - \theta_{init}) \left\| \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} (\hat{r}_{k_2}^- - \hat{r}_{k_1}) \right\|_2. \quad (7)$$

Note that $\hat{\delta}_z$ in Equation (7) is the vertical distance (height difference) computed using the ground inclination angle.

We have assumed that $\hat{\theta}_{k_1}$ is relatively accurate and errors in x and y position estimation in the inertial navigation algorithm are small: that is, we have assumed the following is satisfied

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} (\hat{r}_{k_2}^- - \hat{r}_{k_1}) \approx \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} (r_{k_2} - r_{k_1}).$$

With the assumptions we have the following approximation from Equations (5) and (6):

$$\hat{\delta}_z \approx \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} (r_{k_2} - r_{k_1}). \quad (8)$$

Let v_z be the approximation error in Equation (8), we can rewrite Equation (8) as following:

$$\begin{aligned} \hat{\delta}_z &= \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} (r_{k_2} - r_{k_1}) + v_z \\ &= \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} (r_{k_2} - \hat{r}_{k_2}^- + \hat{r}_{k_2}^- - r_{k_1}) + v_z \\ &= \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} (r_{e,k_2} + \hat{r}_{k_2}^- - \hat{r}_{k_1} - r_{e,k_1}) + v_z \end{aligned} \quad (9)$$

where recall that $r_{e,k} = r_k - \hat{r}_k^-$ in Equation (13).

Let $v_\delta = v_z - \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} r_{e,k_1}$, then Equation (9) can be written as follows:

$$\hat{\delta}_z - \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} (\hat{r}_{k_2}^- - \hat{r}_{k_1}) = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} (r_{k_2} - \hat{r}_{k_2}^-) + v_\delta$$

where v_δ represents the approximation errors.

We have the following measurement equation for the Kalman filter in Appendix:

$$\hat{\delta}_z - \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} (\hat{r}_{k_2}^- - \hat{r}_{k_1}) = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} x_{k_2} + v_\delta. \quad (10)$$

This measurement update equation is combined with the zero velocity updating equation in Equation (16). Note that the measurement noise v_δ contains all the approximation errors in the derivation of Equation (10). So an exact analytic formulation for the covariance is not easy to derive. A small positive value is assigned to $E\{v_{\delta,k_2} v'_{\delta,k_2}\}$ in the paper.

The proposed algorithm combined with the inertial navigation algorithms summarized in the following:

```
while (true)
    compute  $\hat{r}_k^-$ ,  $\hat{v}_k^-$  and  $\hat{q}_k^-$ 
    if ( zero velocity interval )
        if ( the start of the zero velocity interval )
             $\hat{\theta}_{ground} = \hat{\theta}_{ground,previous}$ 
        else if ( the end of the zero velocity interval )
            compute  $\hat{\theta}_{ground,previous}$  using Equations (2) and (3)
        end
        zero velocity updating Equation (16) and height compensation
        Equation (10)
        update  $\hat{r}_k$ ,  $\hat{v}_k$  and  $\hat{q}_k$ 
    else
         $\hat{r}_k = \hat{r}_k^-$ ,  $\hat{v}_k = \hat{v}_k^-$  and  $\hat{q}_k = \hat{q}_k^-$ 
    end
    k = k + 1
end
```

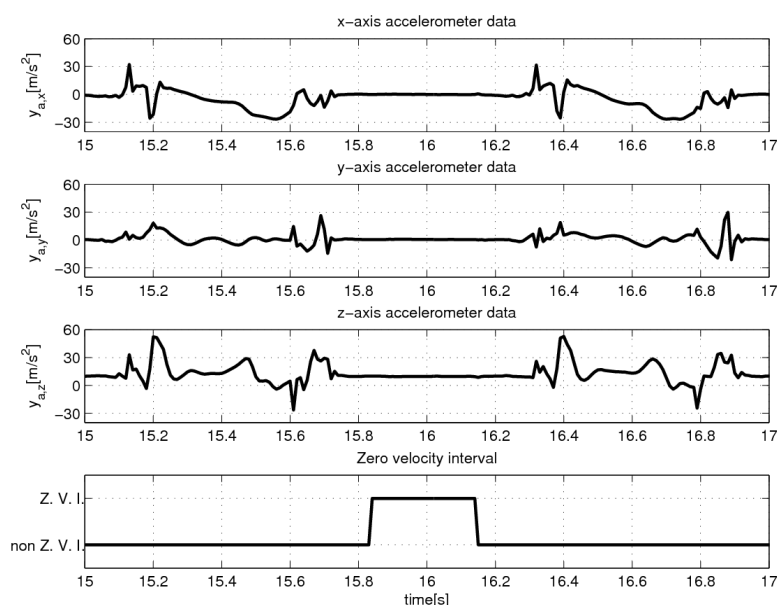
4. Experiments

As an inertial sensor unit, XSens MTi28A53G25 is used, whose specifications are given in Table 1.

In Figure 4, typical accelerometer data (y_a) are given. The accelerometer output from XSens MTi28A53G25 is a low pass filtered signal with the bandwidth 30 Hz (see Table 1). The full scale range of the accelerometer is 50 m/s² and note that there is saturation in $y_{a,z}$, which may cause large z axis velocity and position errors. The detected zero velocity interval is also given.

Table 1. Specifications of XSens MTi28A53G25.

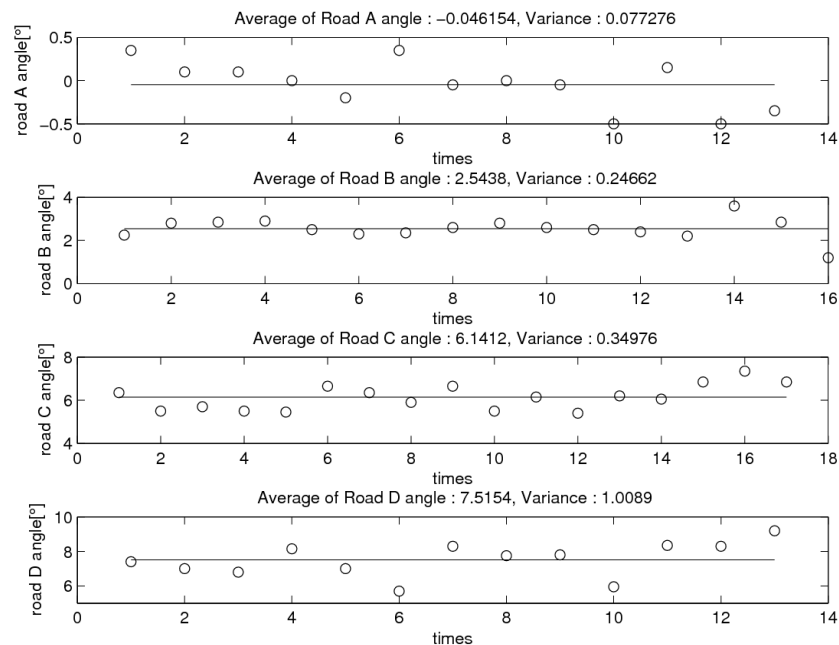
	rate of turn	acceleration	magnetic field
full scale	$\pm 1,500^\circ/\text{s}$	$\pm 50 \text{ m/s}^2$	$\pm 750 \text{ mGauss}$
bandwidth	40 Hz	30 Hz	10 Hz
bias stability	$20^\circ/\text{h}$	0.02 m/s^2	0.1 mGauss
noise	$0.05^\circ/\text{s}/\sqrt{\text{Hz}}$	$0.002 \text{ m/s}^2/\sqrt{\text{Hz}}$	0.5 mGauss

Figure 4. Accelerometer outputs and zero velocity interval.

To test the proposed algorithm, four roads are selected (see Figure 5). These roads are more than 50 m long and the inclination angles are almost constant. The inclination angle of each road is measured with a digital inclinometer: inclination angles are measured at several points (13–17 points for each road) and the average value is considered as the inclination angle of a road. The results are given in Figure 6.

Figure 5. Four roads (A,B,C,D) with different inclination angles.

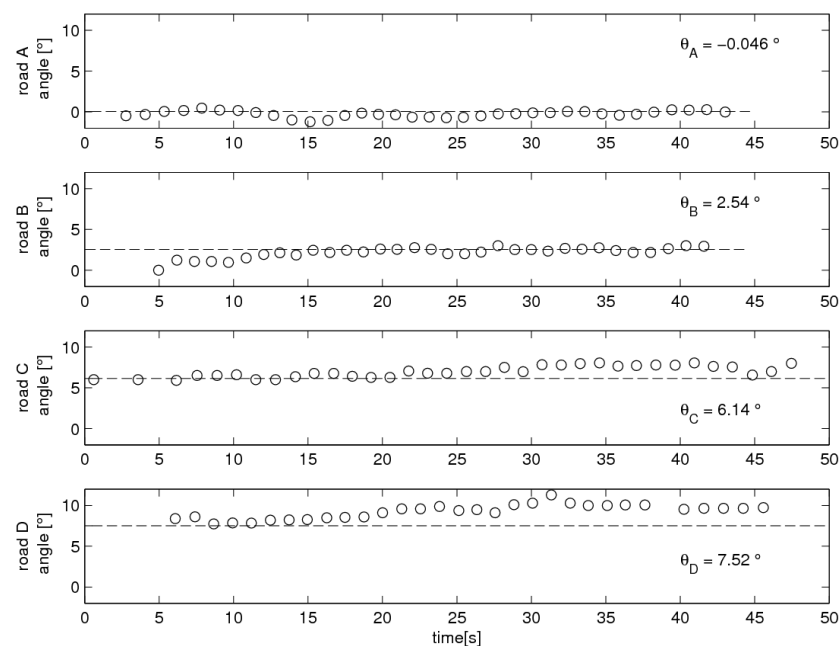
Figure 6. Inclination angle measurement with a digital inclinometer (each measurement is taken from different points along the roads).



In Figure 5, the inclination angles of four roads are -0.046° , 2.54° , 6.14° and 7.52° , respectively.

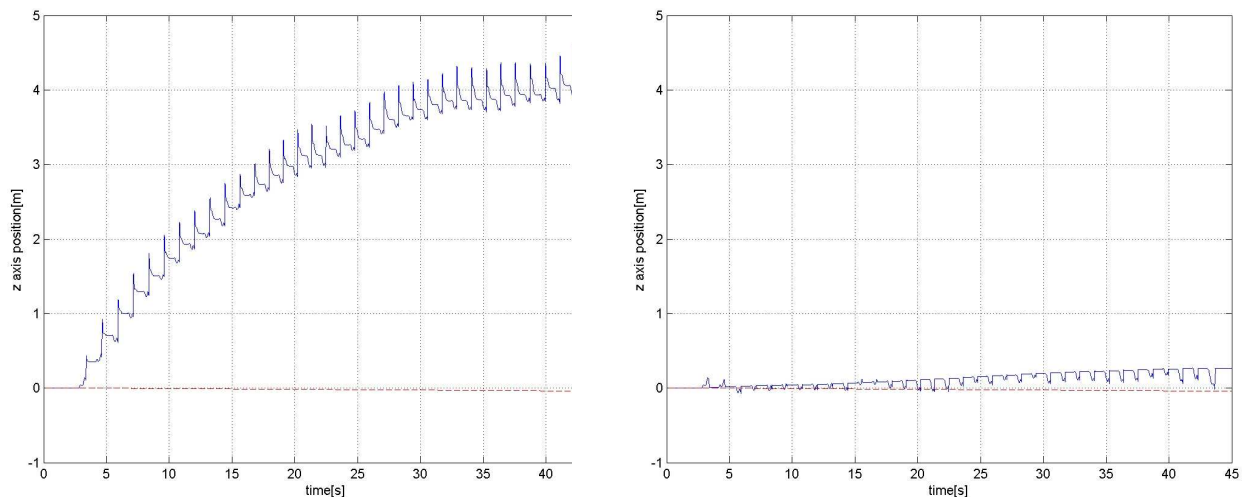
We walked up on each road 50 m and computed θ angle using Equation (3) during the angle measurement interval. In Figure 7, computed $\hat{\theta}_{ground}$ using Equations (2) and (3) is given. Note that each point in the figure corresponds to a computed $\hat{\theta}_{ground}$ for each walking step. For the reference, the road inclination angles are also given. We can see that $\hat{\theta}_{ground}$ is close to the inclination angle and thus $\hat{\theta}_{ground}$ can be used as a road inclination angle estimate.

Figure 7. $\hat{\theta}_{ground}$ estimation for each road (the estimated value at the end of a zero velocity interval).



The z axis position estimation (the third element of r) result for road A (indoor corridor) is given in Figure 8. The left graph in Figure 8 shows the z axis position estimation without the height compensation, where the inertial navigation algorithm with zero velocity updating is used. The straight line in the plot is the estimated actual z axis position, which is computed from the inclination angle of the road and the walking distance (50 m). We can see the error increases rapidly. The error growth depends on many elements such as sensor scaling factor calibration, bias stability, sensor axis alignments and sensor saturation. We only performed simple calibrations. An initial gyroscope bias is estimated by averaging initial 1 minute gyroscope data while the sensor unit is not moving. Also, the accelerometer offset is estimated by rotating the accelerometer 360° and finding the center value. With this simple calibration, the error seems to be large. The right graph in Figure 8 shows that the z axis position is corrected using the proposed height compensation algorithm.

Figure 8. z axis position estimation for road A without (left) and with (right) the proposed height compensation.



The z axis position estimation result for road C (inclination angle of the road is 6.14°) is given in Figure 9, where a person walked up 50 m along the road. The straight line is drawn between 0 and the computed final z axis position ($50 \text{ m} \times \sin(6.14^\circ) = 5.348 \text{ m}$). Note that we measured 50 m using a tape measure on the road and thus 50 m corresponds to $\|r_N - r_1\|_2$, where r_N is the final position and r_1 is the initial position. Without the height compensation, we can see that the z axis position error is large (final z axis position error is 3.61 m). On the other hand, with the height compensation, the z axis position error compensation is greatly reduced (final z axis position error is 0.62 m).

For the same road C, we walked down 50 m along the road and the result is given in Figure 10. It can be seen that without the height compensation, the z axis position error diverges quickly. In the right plot, it can be seen that the z axis position is compensated with the height compensation algorithm.

For four roads, three walking experiments are done. The average z axis position errors are given in Table 2 without and with the compensation algorithm. The true final z position is computed using $\sin(\theta_{\text{ground}}) \times 50 \text{ m}$. We can see that the proposed height compensation algorithm reduces the z axis position error significantly. In Table 2, position errors (with compensation) of Road D seem to be large. We believe this is due to the fact that the road D does not have a smooth surface, which can be verified

from Figure 6. Thus the computed true height (that has been estimated using the estimated slope angle 7.52°) may not be accurate.

Figure 9. z axis position estimation for road C without (left) and with (right) the proposed height compensation (up-walking).

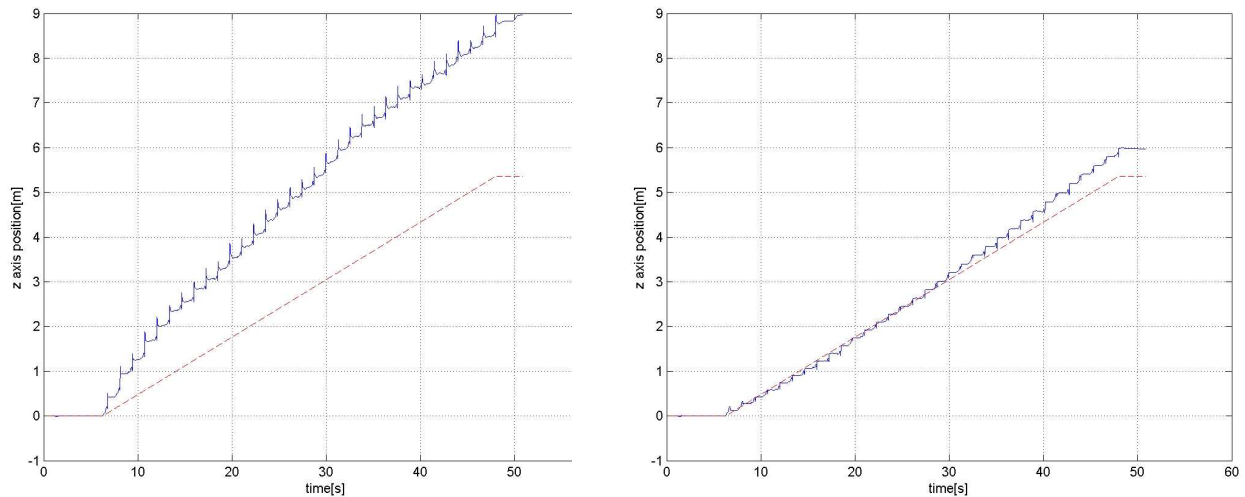
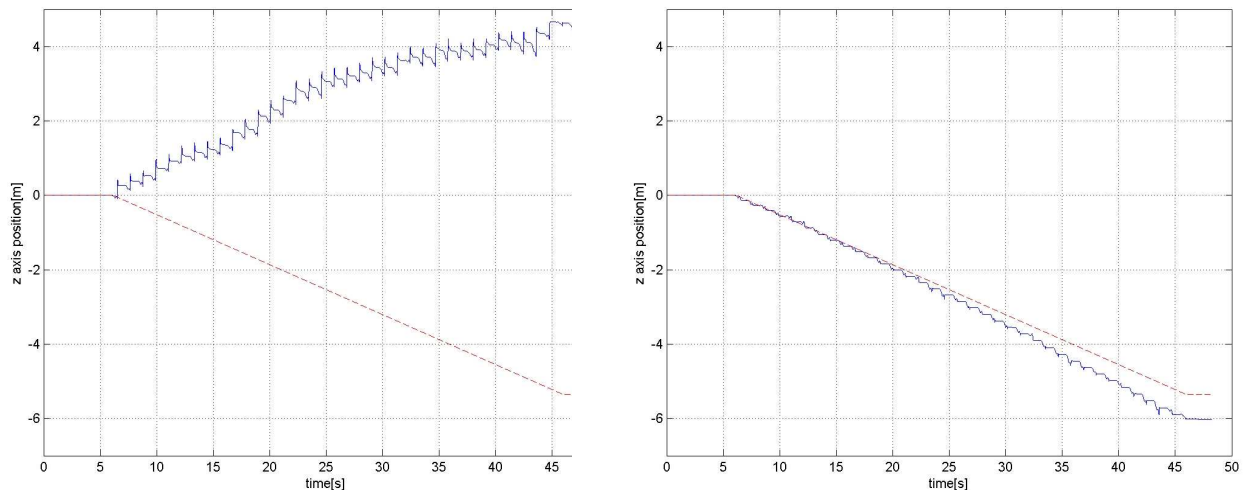


Figure 10. z axis position estimation for road C without (left) and with (right) the proposed height compensation (down-walking).

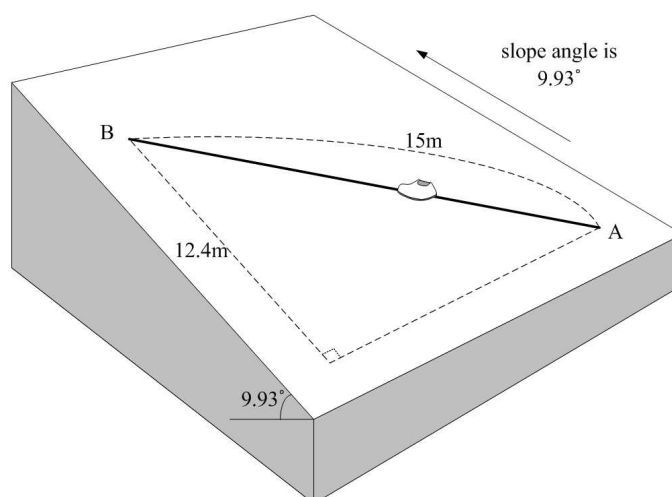


We note although the proposed method reduces the z axis position error growth, the position error divergence cannot be avoided over the long time.

Now instead of walking up and down along the slope, a person walked up and down the slope diagonally. In this experiment, pitch angle ϕ is not zero. We measured road inclination angles along line B in Figure 11 and the average road inclination angle is 7.83° . A person walked up and down 3 times. The z axis position error with the height compensation was 0.45, 0.43, 0.15 m (walking up) and 0.41, 0.30, 0.21 m (walking down). Thus we can see the proposed algorithm is working when ϕ is not zero.

Table 2. z axis position error (average value of 3 experiments).

		position error without compensation (m)	position error with compensation (m)
road A		5.76	0.41
road B	walking up	1.59	0.05
	walking down	3.51	0.30
road C	walking up	1.64	0.40
	walking down	8.94	0.79
road D	walking up	4.62	1.19
	walking down	8.33	2.11

Figure 11. Height compensation experiment while walking up and down the slope diagonally.

5. Conclusions

In pedestrian navigation systems using inertial navigation algorithm, position error tends to diverge sooner or later. To reduce the position error growth, a zero velocity updating algorithm is used. Even with the zero velocity updating algorithm, position error growth could be still large. In particular, the z axis position (height) error growth could be significant.

In this paper, we have proposed a height compensation algorithm. An inclination angle of a road is estimated using foot angle estimation. Using the inclination angle, the height difference of a walking step is estimated. Using this estimation, z axis position in the inertial navigation algorithm is compensated. Through walking test under four different roads (with different inclination angles), the usefulness of the proposed method has been shown. Four different roads (with different inclination angle), 50 m walking test was done. Without the proposed height compensation algorithm, the average z axis position error range was 1.64–8.94 m over 50 m walking. On the other hand, the average error range with the proposed height compensation algorithm was 0.05–2.11 m.

We note that although the proposed method reduces the z axis position error growth, the position error divergence cannot be avoided over the long time. To avoid the divergence problem, external reference such as GPS should be used.

The current algorithm assumes that a person walks up or down on a slope direction. The current algorithm cannot deal with the staircase walking. The future work is to improve the proposed algorithm to cope with various situations such as the staircase walking. One possible solution is to use gait phase information (which can be determined using inertial sensors and force sensors [18]) to determine whether a person is stair climbing or descending.

Acknowledgments

This work was supported by National Research Foundation of Korea Grant funded by the Korean Government (No. 2011-0002770).

Appendix

An indirect Kalman filter for inertial navigation algorithms is introduced in this appendix. The equations are mostly from [10].

Let \hat{q} , \hat{r} , and \hat{v} be the estimates of q , r and v , which are computed from the following equations:

$$\dot{\hat{q}} = \frac{1}{2}\Omega(y_g)\hat{q} \quad (11)$$

where $\Omega(y_g)$ is defined by

$$\Omega(y_g) = \begin{bmatrix} 0 & -y_{g,x} & -y_{g,y} & -y_{g,z} \\ y_{g,x} & 0 & y_{g,z} & -y_{g,y} \\ y_{g,y} & -y_{g,z} & 0 & y_{g,x} \\ y_{g,z} & y_{g,y} & -y_{g,x} & 0 \end{bmatrix}.$$

$$\begin{aligned} \dot{\hat{v}} &= C''(\hat{q})y_a - \tilde{g} \\ \dot{\hat{r}} &= \hat{v}. \end{aligned} \quad (12)$$

where \tilde{g} is the gravitational acceleration vector.

If there are no sensor noises and bias terms in y_a and y_g , \hat{q} , \hat{r} , and \hat{v} should be 100% accurate. For example, if $y_g = \omega_b$, then $\hat{q} = q$ (that is, there is no error in \hat{q}). Due to sensor noises, however, \hat{q} , \hat{r} , and \hat{v} contain errors.

Let q_e , r_e , v_e be errors in \hat{q} , \hat{r} , and \hat{v} , which are defined by

$$\begin{aligned} q_e &\triangleq \hat{q}^* \otimes q \\ r_e &\triangleq r - \hat{r} \\ v_e &\triangleq v - \hat{v} \end{aligned} \quad (13)$$

where \otimes is the quaternion multiplication.

If the error q_e is small, it can be approximated by

$$q_e \approx \begin{bmatrix} 1 \\ \bar{q}_e \end{bmatrix}. \quad (14)$$

The errors are estimated using a Kalman filter and the state for the Kalman filter is given by

$$x \triangleq \begin{bmatrix} \bar{q}_e \\ b_g \\ r_e \\ v_e \\ b_a \end{bmatrix}.$$

The state equation is given by

$$\dot{x}(t) = Ax(t) + B \begin{bmatrix} v_g \\ w_{b_g} \\ v_a \\ w_{b_a} \end{bmatrix} \quad (15)$$

where A and B are given by ($I_3 \in R^{3 \times 3}$ is an identity matrix and $0_3 \in R^{3 \times 3}$ is a zero matrix)

$$A \triangleq \begin{bmatrix} [-y_g \times] & -\frac{1}{2}I_3 & 0_3 & 0_3 & 0_3 \\ 0_3 & 0_3 & 0_3 & 0_3 & 0_3 \\ 0_3 & 0_3 & 0_3 & I_3 & 0_3 \\ -2C'(\hat{q})[y_a \times] & 0_3 & 0_3 & 0_3 & -C'(\hat{q}) \\ 0_3 & 0_3 & 0_3 & 0_3 & 0_3 \end{bmatrix}$$

$$B \triangleq \begin{bmatrix} -0.5 I_3 & I_3 & 0_3 & I_3 \\ 0_3 & I_3 & 0_3 & 0_3 \\ 0_3 & 0_3 & 0_3 & 0_3 \\ 0_3 & 0_3 & -C'(\hat{q}) & 0_3 \\ 0_3 & 0_3 & 0_3 & I_3 \end{bmatrix}.$$

For a vector $p \in R^3$, $[p \times]$ is defined by

$$[p \times] \triangleq \begin{bmatrix} 0 & -p_3 & p_2 \\ p_3 & 0 & -p_1 \\ -p_2 & p_1 & 0 \end{bmatrix}.$$

The noise w_{b_g} and w_{b_a} are process noises for compensation of slowly time-varying gyroscope and accelerometer bias. As usual, we assume all noises in Equation (15) are uncorrelated, zero mean white Gaussian.

When a foot is on the ground and thus is not moving, we can use the fact $v = 0$ in the measurement update of the Kalman filter. The zero velocity interval can be detected using y_a and y_g : if y_a change is small and y_g is small for more than a certain period, we can consider a foot is not moving [15,19]. The zero velocity interval also can be detected using the force sensors installed on a shoe. In this paper, the zero velocity interval is detected using y_a change and the force sensors.

During the zero velocity interval, we can use the fact $v = 0$ in the measurement updating as follows: The velocity measurement y_v is given by

$$\begin{aligned} y_v &= v + v_v \\ &= \hat{v} + v_e + v_v \end{aligned}$$

where v_v is the measurement noise. We assume that the measurement noise v_v is a uncorrelated white Gaussian noise. Let R_v be the covariance of v_v , which is defined by $E\{v_v(t)v_v(s)'\} = R_v\delta(t-s)$.

During the zero velocity interval (that is $v = 0$), we use $0 - \hat{v}$ as an output to the indirect Kalman filter, where the output equation is given by

$$\begin{aligned} -\hat{v} &= v_e + v_v \\ &= \begin{bmatrix} 0 & 0 & 0 & I_3 & 0 \end{bmatrix} x + v_v. \end{aligned} \quad (16)$$

In the zero velocity updating, v_v is not the actual measurement noise since we are not directly measuring v but indirectly estimating $v = 0$. Thus covariance R_v indicates our confidence in the zero velocity algorithm. If we use $R_v = 0_3$, we are assuming that the zero velocity algorithm is 100% correct. We note that there is a slight chance that the zero velocity interval detection is wrong. Thus we assign small positive value to R_v . In the zero velocity updating, the velocity, position and attitude errors are greatly reduced [7,8].

During the zero velocity interval, we also update the heading using y_m . As in [7], heading is only compensated at the end of the zero velocity interval since the attitude is most accurate at that time. For the heading update equation, we used the technique in [20], where y_m only affects heading not pitch and roll. See Equation (21) in [20].

The sampling rate of the inertial sensors is 100 Hz and the appropriate discretized equations are used.

References

1. Llorca, D.F.; Sotelo, M.A.; Parra, I.; Ocana, M.; Bergasa, L.M. Error Analysis in a Stereo Vision-Based Pedestrian Detection Sensor for Collision Avoidance Applications. *Sensors* **2010**, *10*, 3741–3758.
2. Eom, W.; Park, J.; Lee, J. Hazardous Area Navigation with Temporary Beacons. *Int. J. Control Automat. Syst.* **2010**, *8*, 1082–1090.
3. Hazas, M.; Hopper, A. Broadband Ultrasonic Location Systems for Improved Indoor Positioning. *IEEE Trans. Mobile Comput.* **2006**, *5*, 536–547.
4. Kim, S.Y.; Yoon, K.S.; Lee, D.H.; Lee, M.H. The Localization of a Mobile Robot Using a Pseudolite Ultrasonic System and a Dead Reckoning Integrated System. *Int. J. Control Automat. Syst.* **2011**, *9*, 339–347.
5. Shin, S.H.; Park, C.G.; Kim, J.W.; Hong, H.S.; Lee, J.M. Adaptive Step Length Estimation Algorithm Using Low-Cost MEMS Inertial Sensors. In *Proceedings of IEEE Sensors Application Symposium*, San Diego, CA, USA, 6–8 February 2007, pp. 1–5.
6. Feliz, R.; Zalama, E.; Garcia-Bermejo, J.G. Pedestrian Tracking Using Inertial Sensors. *J. Phys. Agents* **2009**, *3*, 35–42.
7. Foxlin, E. Pedestrian Tracking with Shoe-Mounted Inertial Sensors. *IEEE Comput. Graph. Appl.* **2005**, *25*, 38–46.
8. Ojeda, L.; Borenstein, J. Non-GPS Navigation for Security Personnel and First Responders. *J. Navigat.* **2007**, *60*, 391–407.

9. Krach, B.; Roberston, P. Cascaded Estimation Architecture for Integration of Foot-Mounted Inertial Sensors. In *Proceedings of 2008 IEEE/ION Position, Location and Navigation Symposium*, Monterey, CA, USA, 5–8 May 2008; pp. 112–119.
10. Suh, Y.S.; Park, S. Pedestrian Inertial Navigation with Gait Phase Detection Assisted Zero Velocity Updating. In *Proceedings of the 4th International Conference on Autonomous Robots and Agents*, Wellington, New Zealand, 10–12 February 2009; pp. 336–341.
11. Bird, J.; Arden, D. Indoor Navigation with Foot-Mounted Strapdown Inertial Navigation and Magnetic Sensors. *IEEE Wirel. Commun.* **2011**, *18*, 28–35.
12. Titterton, D.H.; Weston, J.L. *Strapdown Inertial Navigation Technology*; IPeter Peregrinus Ltd.: Herts, UK, 1997.
13. Schepers, H.M.; van Asseldonk, E.H.F.; Baten, C.T.; Veltink, P.H. Ambulatory Estimation of Foot Placement During Walking Using Inertial Sensors. *J. Biomech.* **2010**, *43*, 3138–3143.
14. Veltink, P.H.; Slycke, P.; Hemssems, J.; Buschman, R.; Bultstra, G.; Hermens, H. Three Dimensional Inertial Sensing of Foot Movements for Automatic Tuning of a Two-Channel Implantable Drop-Foot Stimulator. *Med. Eng. Phys.* **2003**, *25*, 21–28.
15. Park, S.K.; Suh, Y.S. A Zero Velocity Detection Algorithm Using Inertial Sensors for Pedestrian Navigation Systems. *Sensors* **2010**, *10*, 9163–9178.
16. Savage, P.G. Strapdown Inertial Navigation Integration Algorithm Design Part 1: Attitude Algorithms. *J. Guid. Control Dynam.* **1998**, *21*, 19–28.
17. Savage, P.G. Strapdown Inertial Navigation Integration Algorithm Design Part 2: Velocity and Position Algorithms. *J. Guid. Control Dynam.* **1998**, *21*, 208–221.
18. Pappas, I.P.I.; Popovic, M.R.; Keller, T.; Dietz, V.; Morari, M. A Reliable Gait Phase Detection System. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2001**, *9*, 113–125.
19. Bebek, O.; Suster, M.A.; Rajgopal, S.; Fu, M.J.; Huang, X.; Cauvusoglu, M.C.; Young, D.J.; Mehregany, M.; van den Bogert, A.J.; Mastrangelo, C.H. Personal Navigation via High-Resolution Gait-Corrected Inertial Measurement Units. *IEEE Trans. Instrum. Meas.* **2010**, *59*, 3018–3027.
20. Suh, Y.S. Orientation Estimation Using a Quaternion-Based Indirect Kalman Filter With Adaptive Estimation of External Acceleration. *IEEE Trans. Instrum. Meas.* **2010**, *59*, 3296–3305.