THE JOURNAL OF NAVIGATION (2006), **59**, 135–153.
doi:10.1017/S0373463305003486 Printed in the United Kingdom

MEMS Based Pedestrian Navigation System

Seong Yun Cho

(*Electronics and Telecommunications Research Institute*)
(Email: sycho@etri.re.kr)

Chan Gook Park

(*Seoul National University*)

In this paper we present a micro-electrical mechanical system (MEMS) based pedestrian navigation system (PNS) for seamless positioning. The sub-algorithms for the PNS are developed and the positioning performance is enhanced using the modified receding horizon Kalman finite impulse response filter (MRHKF). The PNS consists of a biaxial accelerometer and a biaxial magnetic compass mounted on a shoe. The PNS detects a step using a novel technique during the stance phase and simultaneously calculates walking information. Step length is estimated using a neural network whose inputs are the walking information. The azimuth is calculated using the magnetic compass, the walking information and the tilt compensation algorithm. Using the proposed sub-algorithms, seamless positioning can be accomplished. However, the magnetic compass based azimuth may have an error that varies according to the surrounding magnetic field. In this paper, the varying error is compensated using the MRHKF filter. Finally, the performance enhanced seamless positioning is achieved, and the performance is verified by experiment.

KEY WORDS

1. Pedestrian Navigation System. 2. Modified RHKF Filter.

1. INTRODUCTION. Recently, the navigational techniques for providing the position information of vehicles have been adopted for computing that of a man. The portable navigation system has been developing based on the E911 (Enhanced 911) implementation requirements that were reported by the Federal Communication Commission (FCC) in 1996. This set out explicitly defined requirements that position information for emergency calls made from mobile phones must be transferred to the 911 public safety answering point (PSAP) with an accuracy of 67% CEP 50 m and 95% CEP 150 m. The portable navigation system has been implemented using GPS, CDMAs pilot signals, AGPS/TDOA, etc. However, these techniques have several limits such as restrictions on the use of GPS signals, many error sources in the CDMA signals, etc.

Another research area for the navigation system is MEMS based pedestrian navigation system (MPNS). In recent years, MEMS technology has allowed production of inexpensive lightweight and small-size inertial sensors with low power consumption. These are all desirable properties for components of a portable navigation system. The quality of the MEMS inertial sensors is, however, conspicuously low. Therefore, a new algorithm is necessary to enhance the performance of the portable navigation system implemented using the MEMS inertial sensors. The technical limit and necessity led the development of algorithms for MPNS. A pedestrian can move only by walking behaviour. Therefore, MPNS is based on the step information and this information can be obtained using inertial sensors. The main idea is that the walking distance is calculated by the estimated step length. Position can be computed by multiplying the walking distance and the azimuth information obtained by using gyros or a magnetic compass. MPNS can be utilized in both indoor and outdoor environments because it is autonomous and not susceptible to external jamming.

Recently, the sub-algorithms for MPNS have been investigated. First, step detection methods using accelerometers have been presented. There are three types of method: peak detection [Jirawimut 2001; Levi 1996; Ladetto 2000; Lee 2001], zero crossing detection [Kappi 2001; Leppakoski 2002] and flat zone detection [Cho 2002; 2003 ION]. In peak and zero crossing detection algorithms, detection miss or over detection can occur because of accelerometer error signal and sensor misalignment. The flat zone detection method is the better method. However, over detection may occasionally occur depending on walking patterns. In this paper, the modified flat zone detection method is presented. The proposed step detection method is robust to the walking velocity, ground inclination, walking pattern, walking environments, etc. Step length estimation methods have been proposed in the MPNS related papers and patents. First, step length is modelled as a linear combination. The linear combination consists of the parameters that have influence on the step length such as walking frequency, variance of the accelerometer signals, etc. [Jirawimut 2001; Levi 1996; Ladetto 2000; Lee 2001; Kappi 2001]. Secondly, walking speed is modelled [Gabaglio 2001; Animian 1995], and third, step length can be calculated by double integration of the accelerometers detecting motion of the foot [Fyfe 1999; Sagawa 2001; Cho 2002]. The first and the second methods have an advantage that the step length can be modelled briefly. However, the nonlinear characteristics cannot be considered. The third method compensates for errors by using the fact that the foot velocity is zero during the stance phase. This method, however, can be affected by the bias of the low-grade accelerometers and the acceleration of gravity. In this paper, a neural network is presented for step length estimation. This method can consider the nonlinear characteristics and is not affected by the accelerometer bias and the acceleration of gravity.

In order to calculate the position, the azimuth information is synchronized with the step information. In the pedestrian environment, the magnetic compass can be utilized usefully. A magnetic compass offers absolute azimuth information by measuring the earth's magnetic field. Therefore, the magnetic compass based azimuth information does not have error increasing with time, unlike gyro based azimuth information. However, it can be influenced by the surrounding magnetic field such as a bridge, buildings, cars, etc. as well as earth's magnetic field. Therefore, the magnetic compass based azimuth information has an error dependent upon the location (Caruso 1997; Cho 2003 IEE).

In order to compensate for varying as well as position error, a particular filter is necessary in the MPNS/GPS integrated system. In general, the Kalman filter is widely

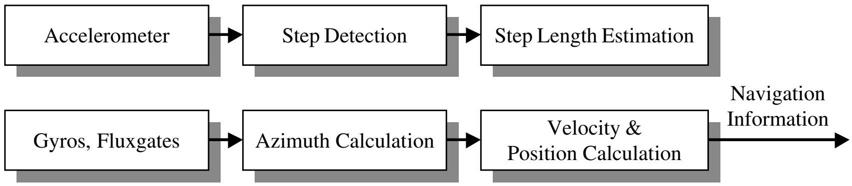


Figure 1. Block diagram of MPNS algorithm.

utilized in the integrated navigation systems (Parkinson 1996; Brown 1997; Farrel 1999; Grewal 2001). When MPNS is used in urban areas, the magnetic compass error varies largely. Unfortunately, the conventional Kalman filter (CKF) cannot estimate the varying error exactly because it has an Infinite Impulse Response (IIR) structure. In this paper a Receding Horizon Kalman Finite Impulse Response (RHKF) filter is adopted to overcome this problem. RHKF filter estimates the state using the measurements only on the current horizon. This filter has a fast estimation property due to the FIR construction (Danyand 1994; Kwon 1999; Ling 1999; Kim 2002). However, MPNS error has nonlinear dynamics and research on the RHKF filter for nonlinear systems is currently insufficient. In this paper, the Modified RHKF (MRHKF) filter is introduced to apply the RHKF filter to nonlinear systems. First, the inverse covariance form of a linearized KF (LKF) is derived because the RHKF filter has a feed-forward structure in the receding horizon. Then it is combined with an Extended KF (EKF) to enhance the convergence characteristics. This filter extends the receding interval to N, the size of the receding horizon of RHKF filter, to reduce the computational burden [Cho 2004]. In order to verify the performance of the MPNS and the MRHKF filter, a pure MPNS and a MPNS/GPS integrated system are implemented and field tests accomplished.

This paper is organized into five Sections. In Section 2, MPNS sub-algorithms for seamless positioning are presented. In Section 3 we present a performance enhancement of MPNS using MRHKF filter. The performance of the proposed methods is verified by experiments in Section 4. Concluding remarks are drawn in the last Section.

2. MEMS BASED PEDESTRIAN NAVIGATION SYSTEM.

Inertial Navigation Systems (INS) for vehicles are a satisfactory technology. A similar approach is, however, difficult to adopt for MPNS. The first problem is the alignment of an IMU. Second, the inherent systematic errors that present in small low-cost inertial sensors quickly accumulate to non-permissible position errors. Such characteristics do not allow one to compute position by double integration of the acceleration. An alternative is to use accelerometer signal pattern rather than its value to count the steps.

Figure 1 shows a main algorithm for pure MPNS. Estimation of the step length is used to compute the distance travelled from the last known or estimated position. When the information on the distance travelled is combined with the azimuth information, the current position can be calculated. As can be seen in Figure 1, operation of the MPNS algorithm can be divided into three parts: step detection, step length estimation, and azimuth calculation. Using physiological models and advanced algorithms, it would be possible to implement PNS using available MEMS sensor technology.

Development of MPNS has been made possible by the various compact, inexpensive sensors. In this paper, the MPNS is implemented by a biaxial accelerometer and a biaxial magnetic compass. This sensor module is then mounted on a shoe. The advanced sub-algorithms used for this sensor module are presented in this Section.

2.1. *Step Detection.* The output of the accelerometer utilized to detect the steps consists of the foot acceleration and the acceleration of gravity. The gravity effect varies due to the ground inclination and the misalignment of the accelerometer even during the stance phase. Unless this effect is eliminated, errors in the step detection may be caused. This effect, however, cannot be removed when the attitude information of the foot is unknown. In this paper, an *acceleration differential* technique is used to eliminate the effect of the acceleration of gravity and to detect the *stance phase* because various information can be obtained in the stance phase such as zero-velocity of the foot, ground inclination, etc. Equation (1) denotes the output of the forward-axis accelerometer attached on a shoe at time t.

$$a_o(t) = a_f(t) + a_g(t) \tag{1}$$

where a_o is the accelerometer output, a_f denotes the foot acceleration, and a_g is the acceleration of gravity.

The foot acceleration is zero during stance phase. Moreover the acceleration of gravity does not vary over this duration. Therefore, the gravity effect can be eliminated through the acceleration differential as follows:

$$\Delta a_o(t) = a_o(t) - a_o(t-1) \cong 0 \tag{2}$$

Therefore, the stance phase can be detected using the condition that the value of the acceleration differential is less than the threshold established by experiments. However, this condition may occur during the swing phase because foot acceleration is about zero at the intersection of acceleration and deceleration of the foot. Therefore, steps may be detected twice in one step. In order to remove this phenomena, the following *sling window summing* technique is used.

$$SWS(t) = \sum_{k=t-N+1}^{t} \Delta a_o(k) \tag{3}$$

where SWS denotes the sliding window summing, and N is the window size that is established to be less than the size of the duration in stance phase.

The SWS still exists in the neighbourhood of zero during the stance phase. However, the sections that correspond to equation (2) during the swing phase have disappeared. Therefore, the probability of over-detection decreases through the following process:

$$\begin{aligned} &\text{i) } Num = 0 \\ &\text{ii) If } |SWS| < \delta_{StP} \text{ then } Num = Num + 1 \\ &\text{iii) If } Num > num_{StP} \text{ then this time is the stance phase.} \end{aligned} \tag{4}$$

where δ_{StP} is the threshold for detection of stance phase and num_{StP} is the number selected during the experimental process.

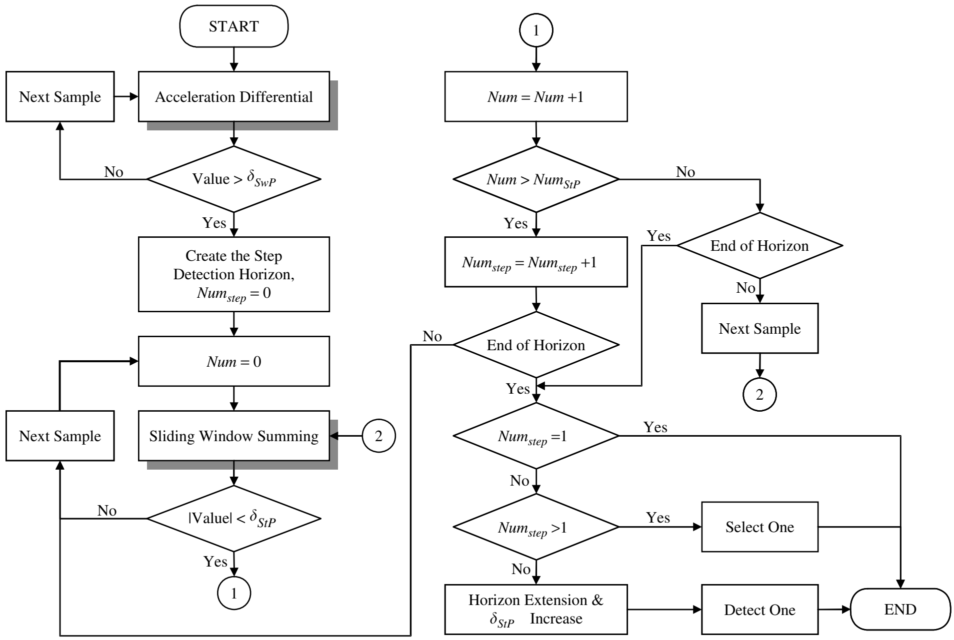


Figure 2. Flowchart of a step detection algorithm.

Figure 2 shows the flowchart of the proposed step detection algorithm. δ_{SwP} is the threshold for swing phase detection. If the value of the acceleration differential exceeds δ_{SwP}, the step detection algorithm is started. Then the step detection horizon is created. In this horizon, just one step must be detected. If the detected number is 2, one step was detected during swing phase. In this case, just one step is selected using the fact that the *Num* in stance phase is generally more than that in swing phase. Therefore, the correct one can be selected. If the detected number is zero, the step detection horizon is extended and the step detection threshold is increased to detect the missed step. Then a step is searched over the horizon. However, these phenomena, where the detected number is not one, rarely occur after applying the acceleration differential and sliding window summing techniques. In conclusion, just one step can be detected in one step using the proposed algorithm.

In order to verify the performance of the proposed step detection algorithm, the accelerometer signal is analyzed as in Figure 3. Figure 3(a) denotes the original accelerometer signals and Figure 3(b) shows the differential value of the accelerometer signals. The bars placed on the figures denote the detected stance phases. It can be seen that the accelerometer signal may not be zero during a stance phases due to the gravity effect. It can be confirmed in Figure 3(b) that the differential values during stance phases are almost zero. The arrow highlights a similar phenomenon occurring during the swing phase. Because of this fact steps can be detected twice within one step. In order to remove this problem, the differential values are summed over the sliding window established previously. The result is that the flat areas close to zero appear only on the stance phases as Figure 3(c).

2.2. *Step Length Estimation.* The step length used to calculate the walking distance in MPNS is the distance a foot moves during a swing phase. If step lengths

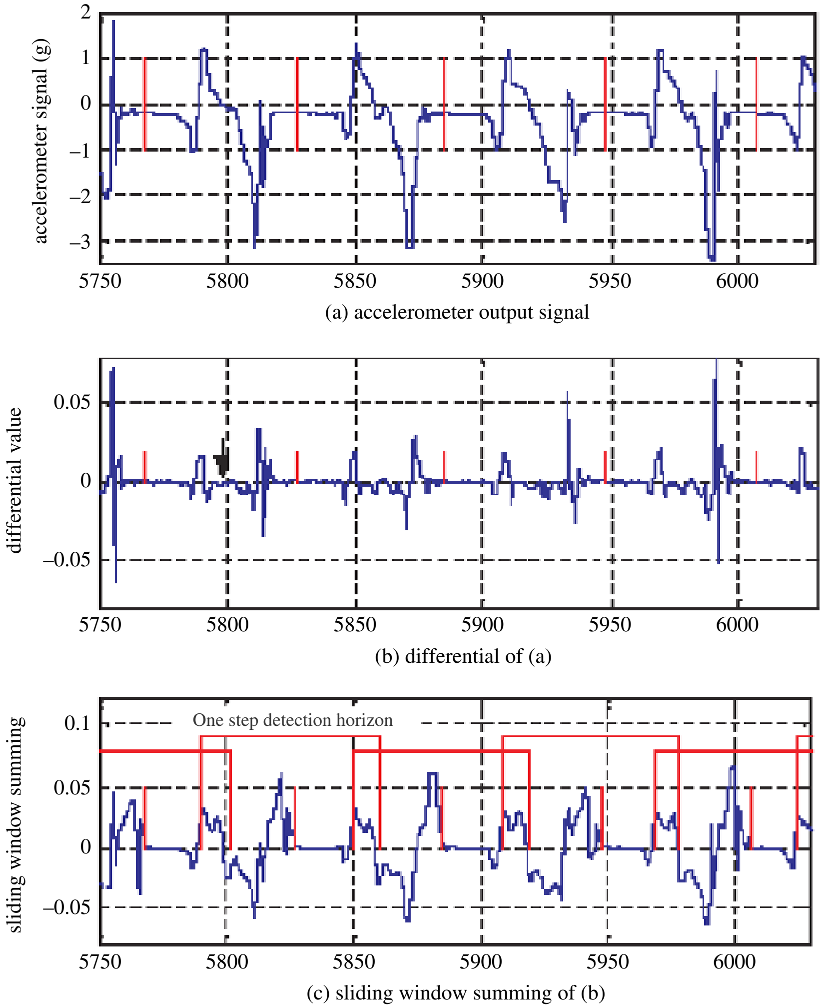


Figure 3. Accelerometer signal analysis.

are constant, the walking distance can be calculated with accuracy. However, step lengths vary continuously according to the walking speed, ground inclination, etc. Unless varied step lengths are considered, the results of MPNS may have large errors.

According to the results of our investigation, step length is influenced by the walking frequency, variance of the accelerometer signals during one step period, ground inclination, etc. That step length is proportional to the walking frequency and the variance of the accelerometer signals is presented in the references [Levi 1996; Ladetto 2000; Lee 2001; Kappi 2001; Cho 2002, 2003 ION]. However, it is confirmed that step length has complex tendency according to the ground inclination as can be seen in Figure 4. The solid lines in Figure 4 are the result of the 1st order curve fitting. It can be seen that step length is proportional to the walking frequency. However, as can be seen in Figures 4(a), (b), and (c), the approximated equation on flat road

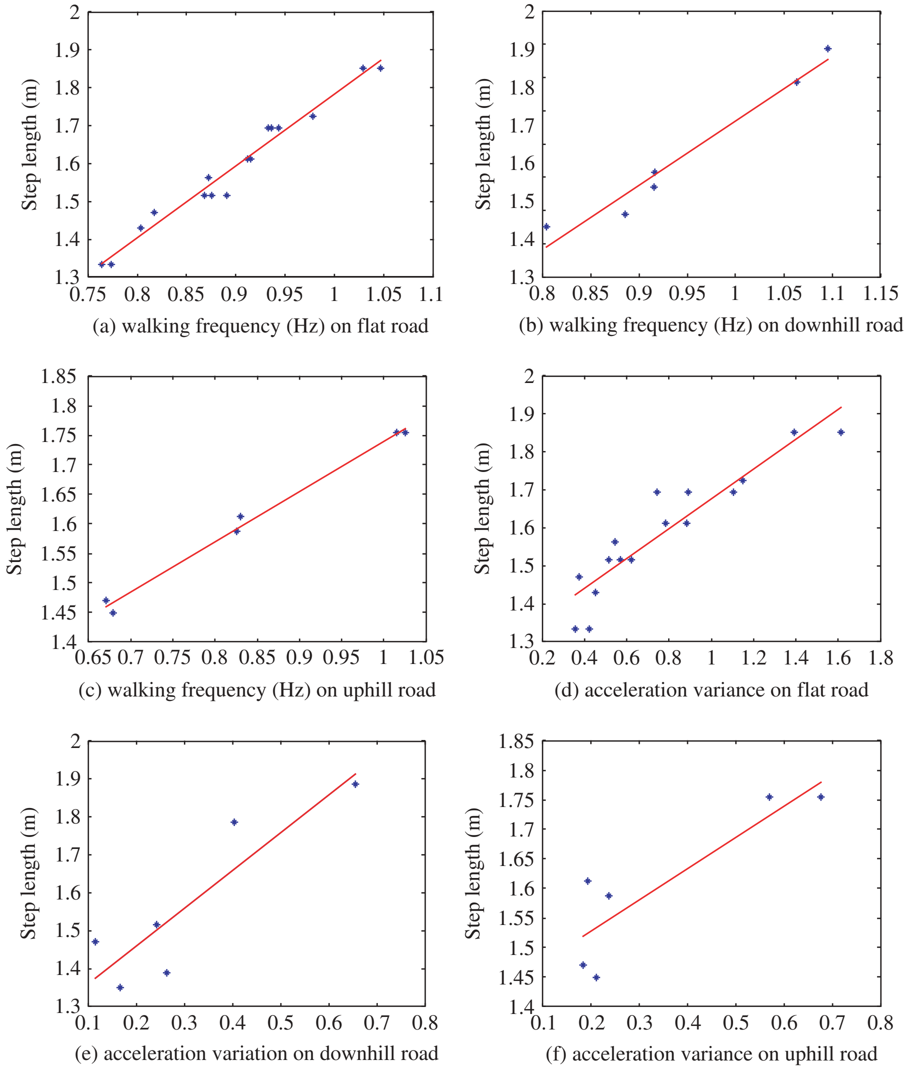


Figure 4. Step length according to walking frequency and acceleration variance on several roads.

is different from that on downhill road and that on uphill road. Step length is proportional to the variance of the accelerometer signals on flat road as can be seen in Figures 4(d), (e), and (f). However, the variance of the accelerometer signals on uphill road cannot be expressed exactly with 1st order approximation. If a pedestrian walks only on a flat road, step length can be modelled as a linear combination as in the previous works [Levi 1996; Ladetto 2000; Kappi 2001]. However, step length cannot be modelled as a linear combination in the case of a sloping environment. The nonlinear problem about the ground inclination can be solved through the heuristic approach such as fuzzy or neural network.

In this paper, the step length is estimated using a neural network shown in Figure 5. The inputs for the neural network consist of the walking frequency, variance of the

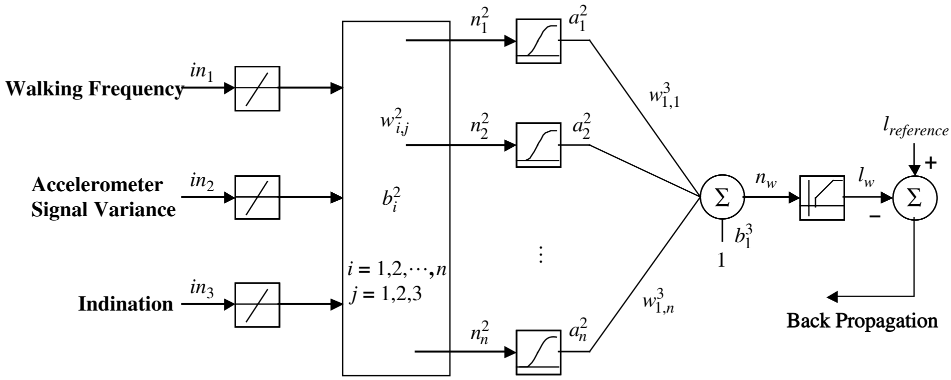


Figure 5. Multilayer neural network for step length estimation.

accelerometer signals, and the ground inclination. The ground inclination can be calculated using the shoe-mounted accelerometer during stance phase. These parameters have influence on the step length and can be obtained as:

$$f(t_k) = 1/(t_k - t_{k-1}) \tag{5a}$$

$$Var(t_k) = \sum_{t=t_k-1}^{t_k} \frac{(a(t) - \bar{a}(t_k))^2}{n} \tag{5b}$$

$$\theta(t_k) = \sin^{-1}(\hat{a}(t_k)/g) \tag{5c}$$

where

$f(t_k)$ walking frequency at t_k; $Var(t_k)$ variance of the accelerometer signals during one step; $\theta(t_k)$ ground inclination; t_k time stamp of detection of the k-th step; $a(t)$ accelerometer signal; $\bar{a}(t_k)$ average of the accelerometer signals during one step; n number of the accelerometer outputs during one step; $\hat{a}(t_k)$ average of the accelerometer signals during stance phase.

In Figure 5, the activation function of the hidden layer is the log-sigmoid function and that of the output layer is selected as follows:

$$l_w = \begin{cases} 0; & n_w < \bar{l}_{lb} \\ n_w; & \bar{l}_{lb} \leqslant n_w \leqslant \bar{l}_{ub} \\ \bar{l}_{ub}; & n_w > \bar{l}_{ub} \end{cases} \tag{6}$$

where, $\bar{l}_{lb}$ is the lower bound for walking and is used to ignore unnecessary motion.

In the design of the neural network, the tuning parameters are the number of the neurons of the hidden layer, the learning rate, the momentum coefficient, and the iteration number. These parameters are tuned by trial and error. The results of training of the neural network are as follows:

$$n = 3, \ \alpha = 0{\cdot}02, \ \gamma = 0{\cdot}0, \ \rho = 0{\cdot}993, \text{ and } num_{ite} = 150 \tag{7}$$

where n is the number of the neurons, α is the constant of proportionality called learning rate, γ is the momentum coefficient, ρ is a parameter to reduce the learning rate as the learning progresses, and num_{ite} denotes the iteration number. And the weight learning of the layers is as follows:

$$w^{3}_{1,k}(t+1) = w^{3}_{1,k}(t) + \gamma \Delta w^{3}_{1,k}(t) + (1-\gamma) 2\alpha\varepsilon a^{2}_{k} \tag{8}$$

$$w^{2}_{i,j}(t+1) = w^{2}_{i,j}(t) + \gamma \Delta w^{2}_{i,j}(t) + (1-\gamma) 2(\rho\alpha)\varepsilon w^{3}_{1,i} e^{-n} a^{2}_{i} in_{j} \tag{9}$$

where $\varepsilon = l_{reference} - l_{w}$. The output layer is learned using (8) and the hidden layer is learned using (9).

Figure 6 shows the result of training the neural network according to walking velocity. The blue line is the reference step length and the red line denotes the estimated step length after learning. As can be seen in this figure, the neural network estimates well the step length variations due to walking velocity.

2.3. *Azimuth Calculation.* The walking direction in MPNS is needed to calculate the walking path. It is assumed that the azimuth is calculated using a biaxial magnetic compass. The magnetic compass measures the earth magnetic field and then calculates the azimuth. Therefore, it has bounded error unlike a gyro. However, the error may include a bias error caused by the surrounding magnetic field and the tilt error generated by the inclination. Generally, the 3-axis magnetic compass and inclinometers are used in order to compensate for the tile error. In this paper, however, a biaxial magnetic compass, a biaxial accelerometer, and the tilt compensation algorithm for 2-axis magnetic compass are used [Cho 2003 IEE] because the 3-axis magnetic compass is too bulky.

When the inclination of the magnetic compass is calculated using the shoe-mounted accelerometer, the acceleration data during the stance phase is utilized. Therefore, the inclination and the azimuth are calculated at the point of time when the step is detected. Figure 7 shows the outputs of the biaxial magnetic compass. As can be seen in this figure, the output of the magnetic compass varies during walking. However, the data during stance phase is stationary as can be seen in Figure 8. Therefore, the stable azimuth information can be calculated using the proposed skill [Cho 2003 ION].

3. PERFORMANCE ENHANCEMENT USING MRHKF FILTER. The error of MPNS increases with time because of the DR construction. But the rate of increase of error in MPNS is lower than that in INS. Figure 9 shows the error propagations of INS and MPNS. It is assumed that the bias of low-cost accelerometer is 10[mg], the walking frequency is 2[Hz], and the estimation error of step length is 0·05[m] in case 1, 0·1[m] in case 2, and 0·2[m] in case 3. The assumed values may occur in real applications. As can be seen in Figure 9, the position error caused by the step length error in MPNS is less than that caused by the bias errors of the inertial sensors in INS. However, the error of MPNS increases with time because of the DR construction. In order to restrict the error increase, MPNS is integrated with GPS using a proper filter.

The magnetic compass used in MPNS measures the surrounding magnetic field as well as earth's magnetic field. From this effect, the azimuth solution obtained from the magnetic compass may have an error. This error can be estimated by a proper filter, such as Kalman filter, in an MPNS/GPS integrated system. However, the error

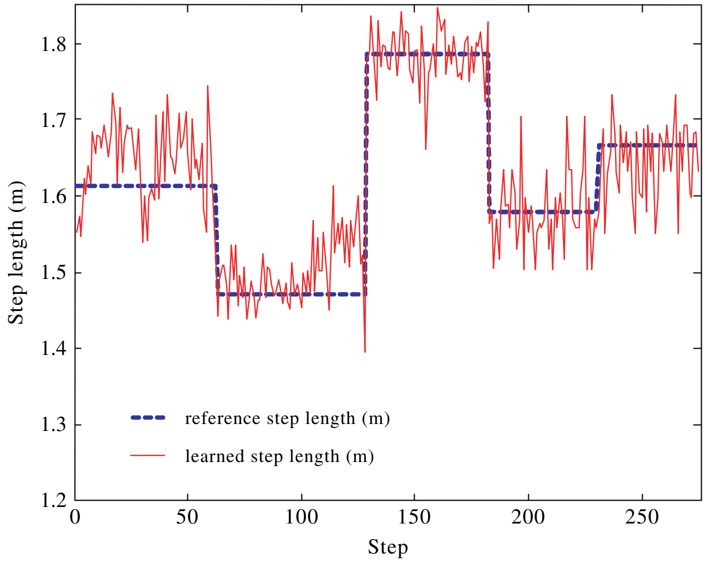


Figure 6. Result of training of the neural network according to the walking velocity.

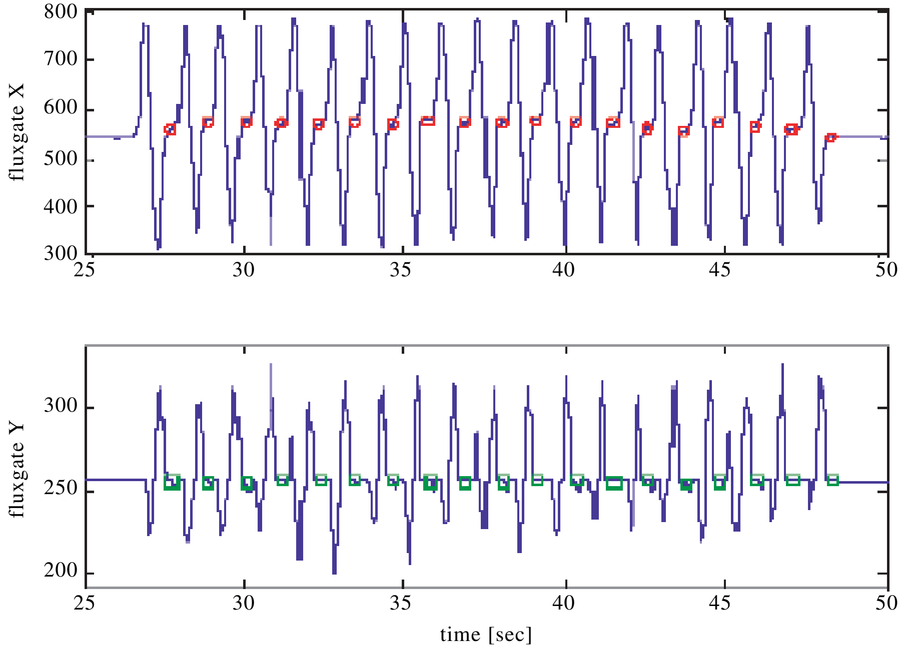


Figure 7. Magnetic compass signals.

may suddenly change based on the surroundings. A conventional Kalman filter cannot estimate this varying error quickly because of its IIR construction. In order to overcome this problem, the receding horizon Kalman FIR filter is adopted in this paper. An RHKF filter can estimate the varying error of the magnetic compass well

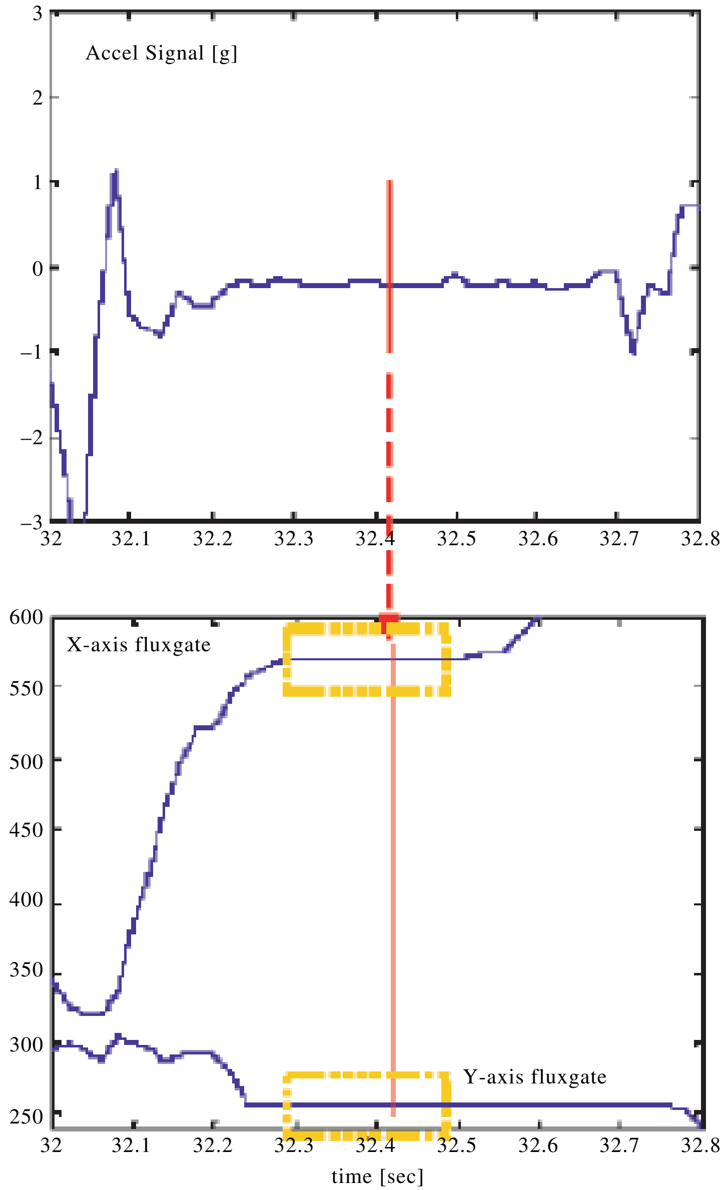


Figure 8. Stable magnetic compass signals during the stance phase.

using the current finite measurements because RHKF filter has a fast estimation property [Kwon 1999; Kim 2002]. In an MPNS/GPS integrated system, the error model of MPNS has nonlinear dynamics. To apply RHKF filter into MPNS/GPS, the modified RHKF filter is utilized [Cho 2004]. MRHKF filter has a hybrid filter construction as can be seen in Figure 10. The MRHKF filter has two advantages. First, the computational burden is decreased because this filter utilizes the

Table 1. Stepwise MRHKF filter.

$t_0 \leqslant t_k \leqslant t_N$ RHKF Filter, (let: $k_i = k-N+i$)	
• Initialization	$x_0^* = \hat{x}_0,\ \Omega_{k-N}=0,\ \hat{\xi}_{k-N}=0$
• Time Propagation ($1 \leqslant i \leqslant N$)	
– Initialization	$\psi_{k_i-1} = F_{k_i-1}^{-T}\Omega_{k_i-1}F_{k_i-1}^{-1}$
	$\Gamma_{k_i-1} = \psi_{k_i-1}G_{k_i-1}(Q_{k_i-1}^{-1} + G_{k_i-1}^T\psi_{k_i-1}G_{k_i-1})^{-1}$
	$x_{k_i}^* = f(x_{k_i-1}^*)$
– Inverse Covariance	$\Omega_{k_i}^- = (I-\Gamma_{k_i-1}G_{k_i-1}^T)\psi_{k_i-1}$
– Pseudo State	$\hat{\xi}_{k_i}^- = (I-\Gamma_{k_i-1}G_{k_i-1}^T)F_{k_i-1}^{-T}\hat{\xi}_{k_i-1} + (I-\Gamma_{k_i-1}G_{k_i-1}^T)F_{k_i-1}^{-T}\Omega_{k_i-1}(F_{k_i-1}^{-1}f(x_{k_i-1}^*) - x_{k_i-1}^*)$
• Measurement Update	
– Inverse Covariance	$\Omega_{k_i} = \Omega_{k_i}^- + H_{k_i}^T R_{k_i}^{-1} H_{k_i}$
– Pseudo State	$\hat{\xi}_{k_i} = \hat{\xi}_{k_i}^- + H_{k_i}^T R_{k_i}^{-1}(z_{k_i} + H_{k_i}x_{k_i}^*)$
$t_{N+1} \leqslant t_k \leqslant t_{2N}$: EKF	
• Initialization ($t_k = t_N$)	$P_N = \Omega_N^{-1},\ x_N^* = \hat{x}_N = \Omega_N^{-1}\hat{\xi}_N$

measurements only twice, while RHKF filter uses the measurements N times, the size of the horizon. Second, the convergence characteristic is enhanced. The summary of the MRHKF filter is shown in Table 1 where a new variable $k_i = k-N+i$ is defined for simplicity.

In order to analyze the performance of the MRHKF filter, simulation is carried out. The condition for simulation is as follows: the total step number is 244; the estimated step lengths and the azimuth information calculated using a magnetic compass have errors. The step length bias is constant and the azimuth error varies with time as follows:

$$\begin{cases} error_k = error_{k-1} - 0{\cdot}3°, & 50 \leqslant k < 80 \\ error_k = error_{k-1} + 0{\cdot}3°, & 110 \leqslant k < 140 \\ error_k = 15°, & 180 \leqslant k < 200 \\ error_k = 5°, & otherwise \end{cases} \tag{10}$$

The capability of estimating the azimuth error is analyzed in EKF, and MRHKF filter. Figure 11 shows the result of the simulation. When the error varies with time, EKF cannot estimate the error exactly in a short time due to the IIR structure. Thus the estimation error is large as can be seen in Figure 11(b). This error increases position error as seen in figure 11(a). The estimation error can make the position error diverge when the GPS signal is not available. On the other hand, the MRHKF filter can estimate the varying error comparatively well because of the FIR structure and the fast estimation property. The computational burden is just twice as much as EKF. Therefore, it is confirmed that MRHKF filter has good performance. The results of the simulation are summarized in Table 2.

4. FIELD TEST.

In this Section, a field test was conducted to verify the performance of the proposed algorithm. First, a PNS sensor module was implemented

Table 2. Estimation error.

	Position [m]		Azimuth Error [deg]		Step Length Bias [m]	
	mean	s. d.	mean	s. d.	mean	s. d.
EKF	2·3001	1·2699	1·3974	5·1488	0·0077	0·0458
MRHKF filter	1·2181	0·4783	0·3424	1·5218	0·0034	0·0521

using a low-cost MEMS-type accelerometer. Second, we walked on the appropriate trajectory for pre-learning of the neural network. Then the field test was conducted on the trajectory of length 3,730 m.

The implemented system consists of a sensor module and a navigation computer module. The sensor module is attached on the user's right or left shoe and the navigation computer module is implemented as a handheld type. The sensors for measuring the body dynamics are included in the sensor module and a GPS receiver is mounted on the navigation computer module which is connected with a PDA or notebook. The components of the sensor module are a biaxial accelerometer (ADXL202E from Analog Devices), a biaxial magnetic compass (TMC3000NF from Tokin), an 8-bit microcontroller (8 MHz AVR Atmega163 from Atmel), a RF transmitter (BIM-418-F) and other electrical parts. The navigation computer module includes a RF receiver, a GPS receiver (Swift-b2 from Axiom), an 8-bit micro-controller (4 MHz AVR Atmega161 from Atmel), and a notebook or PDA. The implemented sensor module is shown in Figure 12. The components of the system are of small-size and low-cost. These factors are important for PNS implementation. The sensor signal is converted to digital data and then transferred to the navigation computer module periodically through the RF module that uses a frequency band-width of 418 MHz and a baud rate of 19,200 bps.

The x-axes of the sensors (accelerometer and magnetic compass) are oriented along the foot's forward direction and the y-axes are perpendicular to the x-axis horizontally. The accelerometer measures the foot dynamics for detecting steps and the acceleration of gravity for calculating ground inclination. The inclination information is utilized to compensate the tilt error of the magnetic compass and to estimate the step length. Azimuth information is calculated using the compensated magnetic compass data. The raw sensor data is logged in the memory of the navigation computer in order to investigate it.

The circular road of the Seoul National University, Korea, was selected for the route of the walking test because there was an accurate digital map of the road available. The route consists of flat, uphill and downhill roads, urban and rural areas, etc. The total walking distance was 3,730 m. A man walked a turn of the route with 2,299 steps. In this paper, two steps are assumed to be one step because the proposed algorithm detects just right or left steps. The number of the detected steps using the proposed step detection algorithm was 2,299 exactly. The step lengths were estimated using the proposed neural network. Figure 13 shows the accumulated distance error every 100 steps. When the step length was calculated using the neural network, the mean error was 1·9573 m and the standard deviation was 2·9989 m. In the case of the linear combination, the mean error was 4·4892 m and the standard deviation was 6·4783 m. From these results, it can be confirmed that the step length estimated using

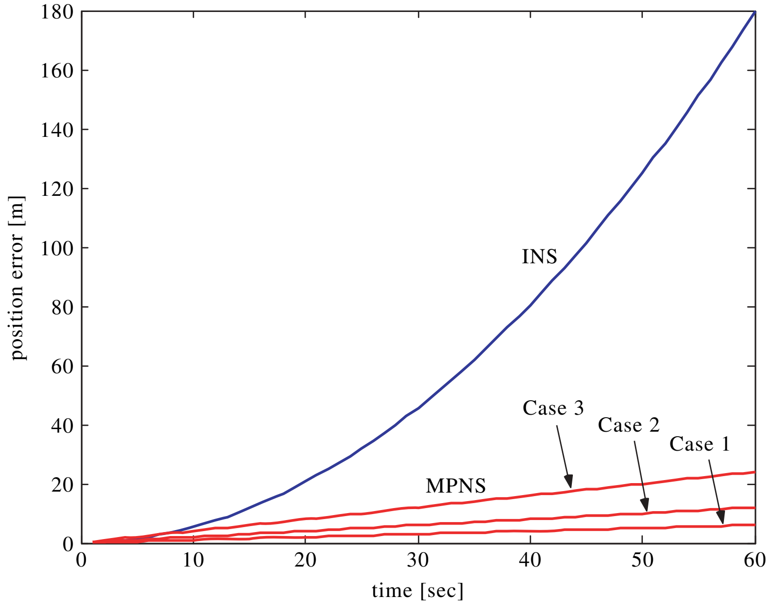


Figure 9. Error propagation of INS vs. MPNS.

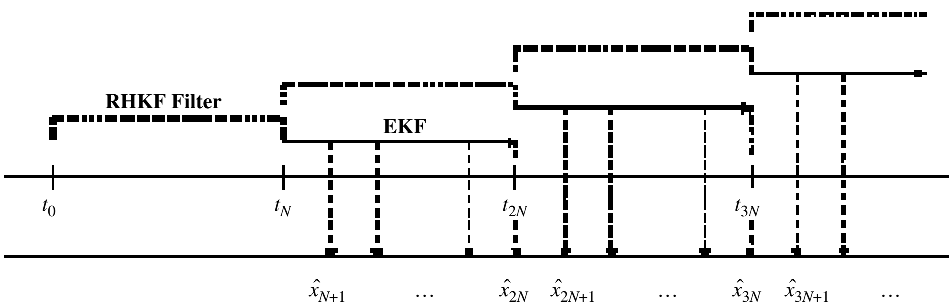


Figure 10. Concept of the MRHKF filter.

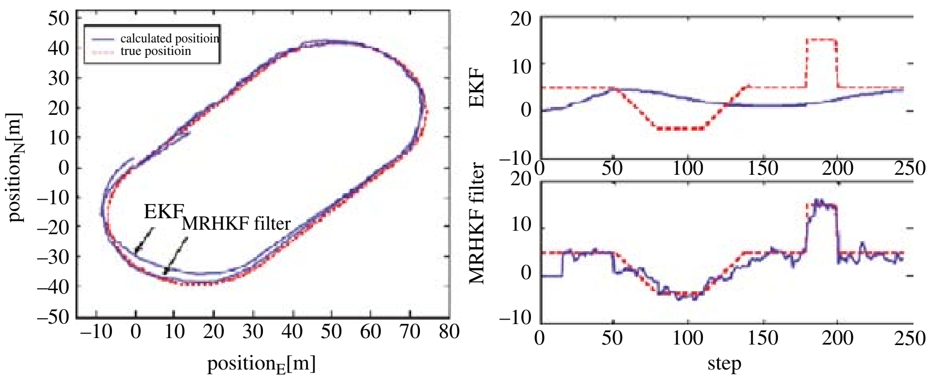


Figure 11. Estimation results.

Figure 12. Implemented sensor module.

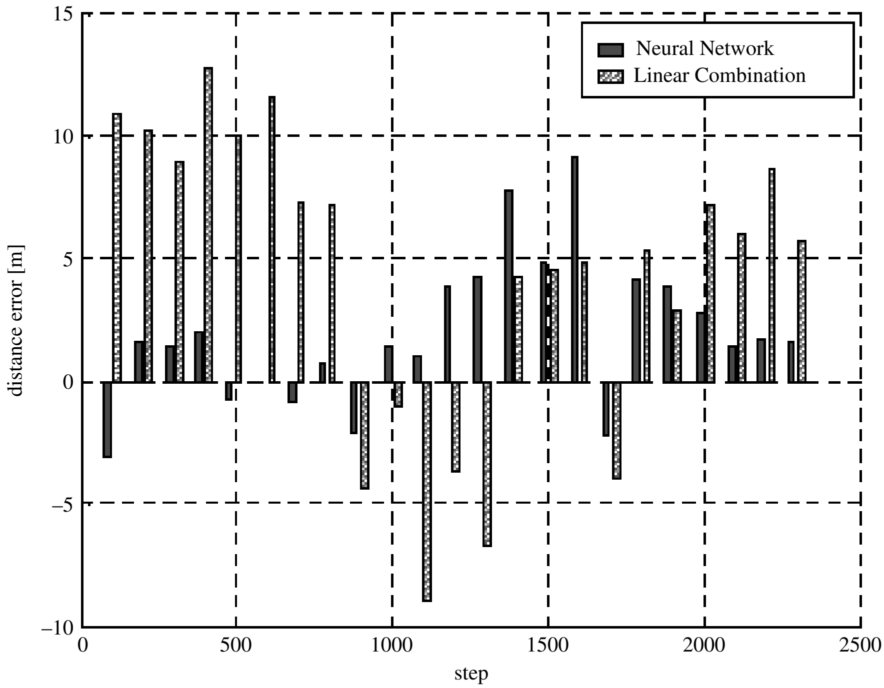


Figure 13. Distance error every 100 steps.

the proposed neural network has better accuracy than that estimated using the linear combination. The reason is that the neural network estimated the step lengths with having regard to the ground inclination as well as the walking frequency. The distance walked was 3,730 m, and the calculated walking distance was 3,788 m using the neural network. The calculation error is about 1·55% of the distance travelled. Therefore, the performance of the proposed algorithm is better than that of the previous works.

In order to compensate MPNS errors, the MPNS was integrated with GPS using two filters: EKF and MRHKF filter. The performance of the MRHKF filter was compared with that of EKF in the MPNS/GPS integrated system. MPNS can be used in both indoor and outdoor environments. When GPS signals are available, the MPNS error is compensated using the filter. After compensation mode, the performance of pure MPNS may be enhanced when GPS signals are not available. However, the performance of pure MPNS may not be improved if the filter estimates the error wrongly in the compensation mode. Figure 14 shows the results of the estimated position. As can be seen the estimated position in pure MPNS has the accurate distance information as mentioned previously and the azimuth errors exist partially. This phenomenon occurs because of the surrounding magnetic field that varies according to location. However, it can be seen that the calculated position deviates from the main trajectory slowly because the estimated step lengths are comparatively exact.

Figure 15(a) and 15(b) show the results of MPNS/GPS integrated system using EKF and MRHKF filter, respectively. It is assumed that the GPS signals are available only in the section A, B, C, and D. GPS signals in these sections are reliable and the MPNS error can be estimated using the GPS data. As can be seen in Figure 15(a) using the EKF, the position errors increase after section A and section C. The reason is that the varying magnetic compass error cannot be estimated exactly in these. On the other hand, the position error in the figure 15(b) where the MRHKF is used, does not diverge after sections A and C. This phenomenon can be explained by the MRHKF filter estimating the varying magnetic compass error exactly during compensation mode – unlike EKF.

Figure 16 shows the estimated magnetic compass bias in section A, B, C, and D. The magnetic compass errors estimated at the end of section A, B, and C by EKF are $-1{\cdot}57$, $3{\cdot}06$, and $-2{\cdot}05$, respectively. The errors estimated by the MRHKF filter are $15{\cdot}46$, $2{\cdot}73$, and $-7{\cdot}39$, respectively. The error estimated at the end of the section B by EKF is similar to that by the MRHKF filter. However, the errors estimated at the end of the section A and C by EKF are different from that by MRHKF filter. As seen in Figure 16(a), the estimated errors at the start of section B and section D by EKF are similar that at the end of section A and section C by MRHKF filter. Moreover, it can be seen in Figure 15(a) that the position error after section A and C in EKF increases unlike that with the MRHKF filter in 15(b). From these phenomena, it can be confirmed that there are errors in the estimated errors at the end of the section A and C in EKF. As can be seen in figure 16(b), the magnetic compass error varied at the midterm of section A and C. The MRHKF filter can estimate the errors at section A and C exactly, while EKF cannot.

5. CONCLUDING REMARKS.

In this paper, sub-algorithms for MEMS based pedestrian navigation system are presented. The proposed step detection algorithm can be implemented using just one shoe-mounted accelerometer. The result of the detection algorithm is robust in walking conditions such as walking velocity, walking type, inclination, etc. Step length is estimated using a neural network. This method considers the ground inclination that was not considered in previous works. The proposed neural network can estimate the step lengths of a pedestrian well irrespective of walking frequency, inclination, etc. with accuracy of 98% of distance travelled. The azimuth can be calculated exactly using a biaxial magnetic compass and a biaxial accelerometer even during walking. An error compensation for MPNS was treated. The azimuth error of MPNS containing magnetic compass varies according to the surrounding magnetic field. In order to compensate this error, MPNS is integrated with GPS using a particular filter. The conventional Kalman filter cannot estimate the varying error successfully due to the IIR construction. In this paper, MRHKF filter, which is able to estimate the varying error accurately due to the FIR structure and the fast estimation property, is applied to MPNS/GPS integrated system. The performance of the proposed sub-algorithms and the MRHKF filter was verified with field tests. The results show that the seamless positioning can be accomplished using the MPNS and that performance enhanced seamless positioning can be achieved using the MPNS with the MRHKF filter.

REFERENCES

Animian, K., Ph. Robert, Jequier, E., and Schutz, Y. (1995). Estimation of Speed and Incline of Walking Using Neural Network. *IEEE Trans. Instrumentation and Measurement*.

Brown, R. G. and Hwang, P. Y. (1997). *Introduction to Random Signals and Applied Kalman Filtering*. Wiley.

Caruso, M. J. (1997). Applications of Magnetoresistive Sensors in Navigation Systems. *Sensors and Actuators 1997*.

Cho, S. Y. (2004). Design of a Pedestrian Navigation System and the Error Compensation Using RHEKF Filter. Ph.D. Dissertation, Department of Control and Instrumentation Engineering, Kwangwoon University, Korea.

Cho, S. Y. and Park, C. G. (2003). Tilt compensation algorithm for 2-axis magnetic compass. *IEE Electronics Letters*, **39**.

Cho, S. Y., Park, C. G., and Jee, G. I. (2002). Measurement System of Walking Distance Using Low-Cost Accelerometers. *Proc, 4th ASCC*.

Cho, S. Y., Park, C. G., and Lee, J. G. (2003). A Personal Navigation System Using Low-Cost MEMS/GPS/Fluxgate. *ION 59th annual meeting*.

Danyand, L. and Xuanhuang, L. (1994). Optimal State Estimation without the Requirement of A Priori Statistics Information of the Initial State. *IEEE Trans. AC*. **39**.

Farrel, J. A. and Barth, M. (1999). *The Global Positioning System and Inertial Navigation*. McGraw-Hill.

Fyfe, K. R. (1999). Motion Analysis System. *United State Patent*, No. 5,955,667.

Gabaglio, V. (2001). Centralized Kalman Filter for Augmented GPS Pedestrian Navigation. *Proc. ION GPS*.

Grewal, M. S., Weill, L. R. and Andrews, A. P. (2001). *Global Positioning Systems, Inertial Navigation, and Integration*. John Wiley & Sons, Inc.

Jirawimut, R., Ptasinski, P., Garaj, V., Cecelja, F., and Balachandran, W. (2001) A Method for Dead Reckoning Parameter Correction in Pedestrian Navigation System. *Proc. 18th IEEE Instrumentation and Measurement Technology Conf*.

Kappi, J., Collin, J., Saarinen, J., and Syrjarinne, J. (2001). MEMS-IMU Based Pedestrian Navigator for Handheld Devices. *Proc. ION GPS*.

Kim, P. S. (2002). Separate-Bias Estimation Scheme with Diversely Behaved Biases. *IEEE Trans. AES*. **38**.

Kwon, W. H., Kim, P. S. and Park, P. G. (1999). A Receding Horizon Kalman FIR Filter for Discrete Time-Invariant Systems. *IEEE, Trans. AC*, **44**.

Ladetto, Q. (2000). On Foot Navigation: Continuous Step Calibration Using Both Complementary Recursive Prediction and Adaptive Kalman Filtering. *Proc. ION GPS*.

Lee, S. W. and Mase, K. (2001). Recognition of Walking Behaviors for Pedestrian Navigation. *Proc. IEEE Int'l Conf. Control Applications*.

Leppakoski, H., Kappi, J., Syrjarinne, J., and Takala, J. (2002). Error Analysis of Step Length Estimation Pedestrian Dead Reckoning. *Proc. ION GPS*.

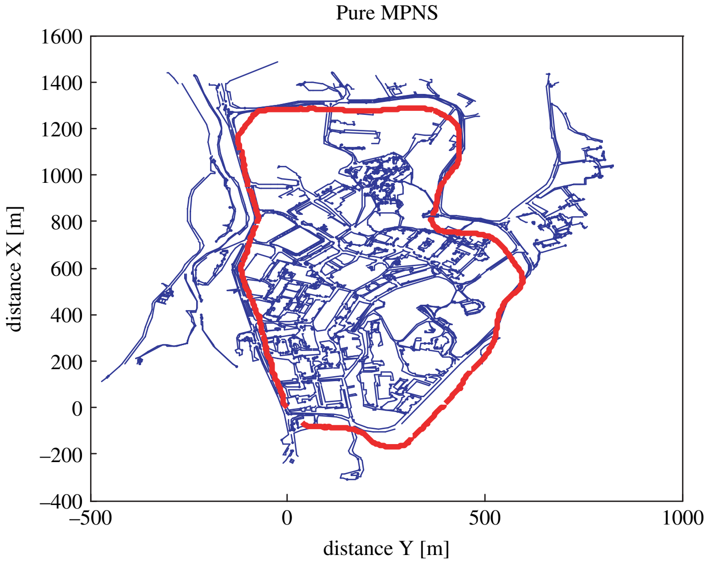
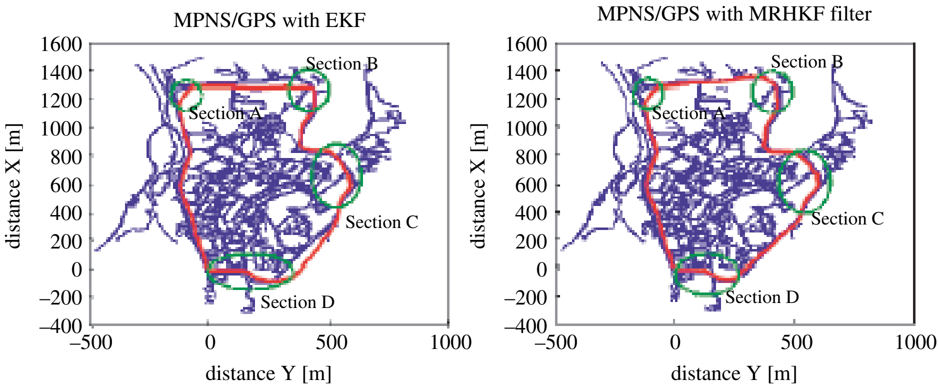
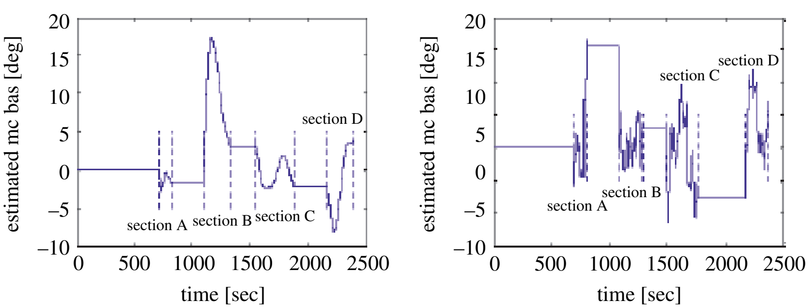


Figure 14. Estimated position by pure MPNS.



(a) MPNS/GPS with EKF (b) MPNS/GPS with MRHKF filter

Figure 15. Estimated position.



(a) EKF (b) MRHKF filter

Figure 16. Estimated magnetic compass error.

Levi, R. W. and Judd, T. (1996). Dead Reckoning Navigational System Using Accelerometer to Measure Foot Impacts. *United States Patent*, No. 5,583,776.

Ling, K. V. and Lim, K. W. (1999). Receding Horizon Recursive State Estimation. *IEEE Trans. AC*, **44**.

Parkinson, B. and Axelad, P. (1996). *Global Positioning System: theory and applications*. American Institute of Aeronautics and Astronautics.

Sagawa, K., Susumago, M., and Inooka, M. (2001). Unrestricted Measurement Method of Three-dimensional Walking Distance Utilizing Body Acceleration and Terrestrial Magnetism. *Proc. ICCAS*.