A Kalman/Particle Filter-Based Position and Orientation Estimation Method Using a Position Sensor/Inertial Measurement Unit Hybrid System

Seong-hoon Peter Won, *Student Member, IEEE*, Wael William Melek, *Senior Member, IEEE*, and Farid Golnaraghi

***Abstract*—This paper presents a novel methodology that estimates position and orientation using one position sensor and one inertial measurement unit. The proposed method estimates orientation using a particle filter and estimates position and velocity using a Kalman filter (KF). In addition, an expert system is used to correct the angular velocity measurement errors. The experimental results show that the orientation errors using the proposed method are significantly reduced compared to the orientation errors obtained from an extended Kalman filter (EKF) approach. The improved orientation estimation using the proposed method leads to better position estimation accuracy. This paper studies the effects of the number of particles of the proposed filter and position sensor noise on the orientation accuracy. Furthermore, the experimental results show that the orientation of the proposed method converges to the correct orientation even when the initial orientation is completely unknown.**

***Index Terms*—Accelerometer, expert system, gyro, inertial measurement unit (IMU), Kalman filter (KF), orientation, particle filter (PF), position.**

I. Introduction

MOST motion tracking systems are utilized for outdoor applications such as vehicle or missile tracking [1]–[3]. Only recently, motion tracking systems have been researched on various indoor applications such as manufacturing [4], human supporting systems [5], [6], and rehabilitation [7]. Even for applications that require only position estimation, many recent tracking systems use a hybrid hardware consisting of one inertial measurement unit (IMU) and one position sensor to improve the position estimation accuracy versus the use of a stand alone position sensor.

An IMU can be used to determine both position and orientation of an object. An IMU consists of three accelerometers and three gyroscopes (hereafter gyros) that provide three acceleration and three angular velocity components. To calculate position and orientation from IMU measurements, acceleration and angular velocity measurements should be integrated. Since inaccuracies of an IMU such as noise, nonlinearity, bias, and gain error are factored in the integration process, this method is a good position and orientation estimator for a short period of time, but this does not hold true as time elapses.

When an IMU is integrated with a position sensor, the drawbacks of each sensor are compensated, and more accurate state estimation can be obtained. To integrate one position sensor with one IMU, variants of Bayes filter such as Kalman filter (KF) [8], [9] or particle filter (PF) [10] are widely used. A KF is an optimum observer that estimates the states of linear Gaussian state-space models. KF and its variants [4], [11], [12] are the most commonly used filtering techniques to integrate an IMU with a position sensor. When the model is highly nonlinear or the noise is non-Gaussian, PF is more suitable because PF neither requires the state-space model to be linear nor assumes that the noise is Gaussian. In addition, even when the initial states are unknown, they will be converged to the correct values in PF. A PF approximates posteriors with a set of state samples, called particles, instead of assuming that the posteriors are Gaussian at every time step. As the number of particles increases, the approximated posteriors get closer to the true posterior, which offers more accurate estimation at the cost of higher computational complexity [13], [14]. To reduce the computational complexity, the linear Gaussian part of the system can be solved using a KF while the remaining part is solved using a PF [15]. This combination provides improved results even with a fewer number of particles.

To achieve drift-free orientation estimation, multiple position markers can be attached to the tracking object. A multiantenna GPS receiver is utilized to find the orientation of a vehicle in [16] and [17], and multiple ultrasonic markers are used to find orientation in [18]. Multiple position markers are integrated with one IMU to obtain drift-free position and orientation estimation in [19] and [20]. When these methods are used, the position markers should be attached to a rigid body, and the markers should be positioned some distance apart from each other to obtain meaningful orientation measurements. However, this requirement is not feasible in many applications due to small object size or the restrictions of applications. Therefore, a hybrid system consisting of one IMU and one position marker is widely used to estimate position and orientation.

This paper presents a novel position/orientation estimation method that uses one position sensor and one IMU for a general in-door position/orientation tracking application. The proposed method estimates orientation using a PF and estimates the position of each orientation particle using a KF. The simulation results of the proposed method when constant angular

Manuscript received April 30, 2009; revised September 10, 2009. First published September 22, 2009; current version published April 14, 2010.

S. P. Won and W. W. Melek are with the Department of Mechanical Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada (e-mail: shown@engmail.uwaterloo.ca; wmelek@mecheng1.uwaterloo.ca).

F. Golnaraghi is with Simon Fraser University, Burnaby, BC V5A 1S6, Canada (e-mail: mfgolnar@sfu.ca).

Digital Object Identifier 10.1109/TIE.2009.2032431

velocity biases are applied to gyros are presented in [21]. This paper presents experimental results that compare the orientation errors for the proposed method with an EKF approach. In addition, the effects of the number of particles and position measurement noise on the orientation error are studied. The proposed method also offers the advantage of being able to estimate orientation when the initial orientation of the object is completely unknown.

In Section II, an introduction to recursive Bayesian state estimation techniques including KF and PF are provided. Section III describes the proposed method which incorporates PF and KF to estimate position and orientation. In addition, this section presents an angular velocity correction technique using an expert system. Section IV presents the experimental validation of the proposed method. Finally, the conclusions are provided in Section V.

II. Recursive Bayesian Estimation

A. Bayes Filter

For system identification purpose, consider the following dynamic state-space model:

$$x_k = f_k(x_{k-1}, u_{k-1}, v_{k-1}) \tag{1}$$

$$z_k = h_k(x_k, e_k) \tag{2}$$

where subscript k represents the iteration number, x_k is the state, u_{k-1} is the deterministic input, v_{k-1} is the process noise, f_k is a state transition function from time t_{k-1} to time t_k, z_k is the measurements, e_k is the measurement noise, and h_k is a measurement function. The stochastic estimation of the current state (x_k) is presented through the posterior probability density function (pdf) when all the measurement up to the current time instant $(z_{1:k})$ and all the inputs up to the previous time instant $(u_{0:k-1})$ are given. Bayes filter calculates the posterior pdf, $p(x_k \mid z_{1:k}, u_{0:k-1})$, in two steps: 1) prediction and 2) update. Using the Bayes rule and the Markov property, that is if the current state is known, the future state is independent of the past states, the prediction and update step can be formulated as

Prediction:

$$p(x_k \mid z_{1:k-1}, u_{0:k-1}) = \int p(x_k \mid x_{k-1}, u_{k-1}) \cdot p(x_{k-1} \mid z_{1:k-1}, u_{0:k-2}) \cdot dx_{k-1} \tag{3}$$

Update:

$$p(x_k \mid z_{1:k}, u_{0:k-1}) = \frac{p(z_k \mid x_k) \cdot p(x_k \mid z_{1:k-1}, u_{0:k-1})}{p(z_k \mid z_{1:k-1}, u_{0:k-1})} \tag{4}$$

where $p(z_k \mid x_k)$ is the likelihood, $p(x_k \mid z_{1:k-1}, u_{0:k-1})$ is the prior, and $p(z_k \mid z_{1:k-1}, u_{0:k-1})$ is a normalizing factor. In order to construct the posterior pdf, the prior should be available including the initial pdf $p(x_0)$.

A Bayes filter requires integration over the state space, which is often impossible to calculate analytically. In some cases, the posterior distribution can be calculated such as the case of a linear Gaussian state-space model (i.e., KF). When the analytical computation is not feasible, the posterior density can be approximated by an estimator such as PF.

B. Kalman Filter

The KF presents an optimal solution of a Bayes filter by assuming that the posterior density is Gaussian. In order for the posteriors to be Gaussian at every time step, the following conditions have to be satisfied [22], [23].

1) The initial pdf is Gaussian.
2) $f_k(x_{k-1}, u_{k-1}, v_{k-1})$ is a linear function of x_{k-1}, u_{k-1} with added zero mean Gaussian noise.
3) $h_k(x_k, e_k)$ is a linear function of x_k with added zero mean Gaussian noise.

By using these assumptions, (1) and (2) become

$$x_k = \Phi_k \cdot x_{k-1} + \Gamma_k \cdot u_{k-1} + v_{k-1} \tag{5}$$

$$z_k = H_k \cdot x_k + e_k \tag{6}$$

where Φ_k is the system transition matrix from time t_{k-1} to time t_k, Γ_k is the input matrix, and H_k is the measurement matrix. The KF assumes that the posteriors of the process noise and the measurement noise are zero-mean Gaussian distributions, and the initial state (x_0), the covariance of process noise (Q_k), and the covariance of measurement noise (R_k) are known. In specific terms, the prediction and update equations of KF are as follows:

Prediction:

Predicted state:

$$\tilde{x}_k = \Phi_k \cdot \hat{x}_{k-1} + \Gamma_k \cdot u_{k-1} \tag{7}$$

Prediction covariance:

$$\tilde{P}_k = \Phi_k \cdot \hat{P}_{k-1} \cdot \Phi_k^{\mathrm{T}} + Q_{k-1} \tag{8}$$

Update:

Kalman gain:

$$K_k = \tilde{P}_k \cdot H_k^{\mathrm{T}} \cdot \left[H_k \cdot \tilde{P}_k \cdot H_k^{\mathrm{T}} + R_k \right]^{-1} \tag{9}$$

Estimated covariance:

$$\hat{P}_k = [I - K_k \cdot H_k] \cdot \tilde{P}_k \tag{10}$$

Estimated state:

$$\hat{x}_k = \tilde{x}_k + K_k \cdot (z_k - H_k \cdot \tilde{x}_k). \tag{11}$$

C. Sampling Importance Resampling PF [23]

The PF is a suboptimal solution that approximates the true posterior with a finite number of random state samples with the corresponding normalized weights. Then, the posterior density approximation at time t_k is

$$p(x_k \mid z_{1:k}, u_{0:k-1}) \approx \sum_{i=1}^{N} w_k^i \delta\left(x_k - x_k^i\right) \tag{12}$$

where $\delta(\cdot)$ is the Dirac delta function, N is the number of samples, w_k^i is the normalized weight of the ith particle, and x_k^i is the ith particle.

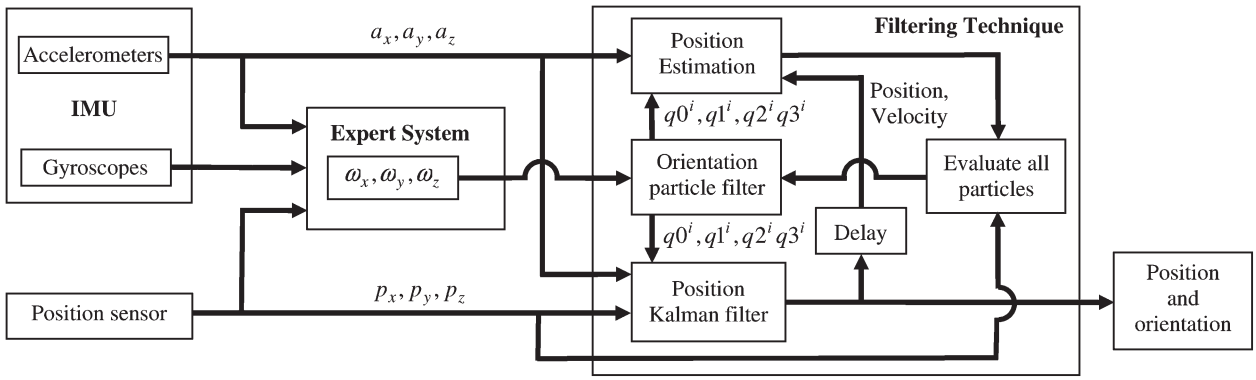


Fig. 1. Outline of the proposed method.

In order to derive the PF state estimation algorithm, consider the posterior density up to time t_k

$$p(x_{0:k} \mid z_{1:k}, u_{0:k-1}) = \frac{p(z_k \mid x_k) \cdot p(x_k \mid x_{k-1}, u_{k-1}) \cdot p(x_{0:k-1} \mid z_{1:k-1}, u_{0:k-2})}{p(z_k \mid z_{1:k-1}, u_{0:k-1})}. \tag{13}$$

Since it is usually difficult to sample from the posterior density, importance sampling technique [24] is used to sample in PF. When the target density (posterior density in this case) can be evaluated at any point but is difficult to sample from, samples can be drawn from a known normalized probability density $[r(x)]$, so-called importance density. To compensate for the difference between the target density and the importance density, normalized weights, which are the ratios of the two densities, are assigned to all particles [22]. The discrete posterior density approximation up to time t_k can be expressed as

$$p(x_{0:k} \mid z_{1:k}, u_{0:k-1}) \approx \sum_{i=1}^{N} w_k^i \delta\left(x_{0:k} - x_{0:k}^i\right). \tag{14}$$

The normalized weight has the following relationship with the target density $p(x)$ and importance density $r(x)$:

$$w_k^i \propto \frac{p\left(x_{0:k}^i | z_{1:k}, u_{0:k-1}\right)}{r\left(x_{0:k}^i | z_{1:k}, u_{0:k-1}\right)} \propto \frac{p\left(z_k | x_k^i\right) \cdot p\left(x_k^i | x_{k-1}^i, u_{k-1}\right) \cdot p\left(x_{0:k-1}^i | z_{1:k-1}, u_{0:k-2}\right)}{r\left(x_{0:k}^i | z_{1:k}, u_{0:k-1}\right)}. \tag{15}$$

The importance density should be chosen so that it can be determined recursively as

$$r(x_{0:k} \mid z_{1:k}, u_{0:k-1}) = r(x_k \mid x_{0:k-1}, z_{1:k}, u_{0:k-1}) \cdot r(x_{0:k-1} \mid z_{1:k-1}, u_{0:k-2}). \tag{16}$$

When the importance density also satisfies the Markov property like the target density, (15) can be written as

$$w_k^i \propto \frac{p\left(z_k | x_k^i\right) \cdot p\left(x_k^i | x_{k-1}^i, u_{k-1}\right) \cdot p\left(x_{0:k-1}^i | z_{1:k-1}, u_{0:k-2}\right)}{r\left(x_k^i | x_{k-1}^i, z_k, u_{k-1}\right) \cdot r\left(x_{0:k-1}^i | z_{1:k-1}, u_{0:k-2}\right)} \propto \frac{p\left(z_k | x_k^i\right) \cdot p\left(x_k^i | x_{k-1}^i, u_{k-1}\right)}{r\left(x_k^i | x_{k-1}^i, z_k, u_{k-1}\right)} \cdot w_{k-1}^i. \tag{17}$$

To simplify (17), the importance density can be chosen from prior as

$$r(x_k \mid x_{k-1}, z_k, u_{k-1}) = p(x_k \mid x_{k-1}, u_{k-1}). \tag{18}$$

Then, (17) can be rewritten as

$$w_k^i \propto w_{k-1}^i \cdot p\left(z_k \mid x_k^i\right). \tag{19}$$

A problem with this type of PF is that only one particle will have high weight (close to unity) and the remaining particles will have negligible weights (almost zero) when k is high. This phenomenon, called degeneracy problem, is undesirable because the weighted particles do not represent the true posterior density. In order to avoid this problem, particles can be resampled based on their weights. Resampling draws more samples from the higher weights and reduces the number of samples from the lower weights. After resampling, all particles are assigned the same weight; thus, the weights at time t_{k-1} are the same $(w_{k-1}^i = 1/N)$. Then, (19) becomes

$$w_k^i \propto p\left(z_k \mid x_k^i\right). \tag{20}$$

In addition, after resampling at time t_k, (12) can be written as

$$p(x_k \mid z_{1:k}, u_{0:k-1}) \approx \frac{1}{N} \sum_{i=1}^{N} \delta\left(x_k - x_k^i\right). \tag{21}$$

Some approaches in the literature propose to calculate the weights based on the "fitness" value [25], [26] or the "evidence" value [27], [28] of each particle to represent the likelihood in (4).

III. Proposed Method

A. Overview

The proposed position/orientation estimation method is shown in Fig. 1. The proposed method calculates the orientation and position states from IMU measurements and measures the position using a position sensor. Based on the measurements from the IMU and position sensor, an expert system is utilized to correct angular velocities. By using the corrected angular velocities (ω_x, ω_y, and ω_z), the rotation matrix from the body frame to the fixed frame is estimated using a PF. The rotation matrix of the ith particle from the body frame to the fixed

frame at time t_k ($^f_bC^i_k$) is represented using a quaternion [29], [30], as in (22), shown at the bottom of the page. In (22), $c\#^i_k$ is the component of the direction cosine matrix of the ith particle at the kth iteration, and $q0^i_k$, $q1^i_k$, $q2^i_k$, and $q3^i_k$ are the four components of a unit quaternion [30]. Since orientation is calculated using a unit quaternion, each orientation particle consists of four states ($x^i_{\mathrm{PF},k} = [\, q0^i_k \quad q1^i_k \quad q2^i_k \quad q3^i_k \,]$) that satisfy

$$q0^{i^2}_k + q1^{i^2}_k + q2^{i^2}_k + q3^{i^2}_k = 1. \tag{23}$$

The quaternion components of each particle can be computed from the angular velocity measurements using

$$\begin{bmatrix} q0^i_{k+1} \\ q1^i_{k+1} \\ q2^i_{k+1} \\ q3^i_{k+1} \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 2 & -\omega_{x,k}\cdot t & -\omega_{y,k}\cdot t & -\omega_{z,k}\cdot t \\ \omega_{x,k}\cdot t & 2 & \omega_{z,k}\cdot t & -\omega_{y,k}\cdot t \\ \omega_{y,k}\cdot t & -\omega_{z,k}\cdot t & 2 & \omega_{x,k}\cdot t \\ \omega_{z,k}\cdot t & \omega_{y,k}\cdot t & -\omega_{x,k}\cdot t & 2 \end{bmatrix} \cdot \begin{bmatrix} q0^i_k \\ q1^i_k \\ q2^i_k \\ q3^i_k \end{bmatrix} \tag{24}$$

where t is the sample time.

The corresponding velocity and position of each orientation particle are estimated using a KF. The motion tracking equations using an IMU with respect to the local fixed frame for short distance navigation are given as [8]

$$^f\dot{v}^i = {}^f_bC^i \cdot {}^bf + {}^fg \tag{25}$$

$$^f\dot{p}^i = {}^fv^i \tag{26}$$

where $^fv^i$ represents the velocity of the ith particle in the fixed frame, ^bf represents the specific force acting on the object in the body frame, g is the local gravity vector, and p is the position of the object. Equation (25) shows that the acceleration measurements ($^bf = [\, a_x \quad a_y \quad a_z \,]^{\mathrm{T}}$), which includes the gravity vector, are multiplied by the rotation matrix to calculate the acceleration in the fixed frame. Thus, when the rotation matrix is inaccurate, the misalignment between the true body frame and the calculated body frame leads to acceleration error in (25), which consequently results in velocity and position errors. In other words, high orientation error can result in high position error. The proposed filter evaluates orientation particles by comparing the position calculation of each particle with the position estimate of each particle from the KF. The particles with lower position difference between the two are assigned higher weights.

B. Position KF

A KF is used to estimate the position of each particle in the proposed method because the state-space model is linear and the noise distribution of many position sensors such as infrared can be considered zero-mean Gaussian. To utilize the KF, the motion equations in (25) and (26) must be rewritten in the form presented in (5). The state vector of the KF of the ith particle at time t_k is defined as (27), shown at the bottom of the page, where $P^i_{k\text{-axis},k}$, $V^i_{k\text{-axis},k}$, and $A^i_{k\text{-axis},k}$ are the position, velocity, and acceleration of the ith particle in each axis using the KF. Then, the system transition matrix of the ith particle Φ^i_k is [8]

$$\Phi^i_k = \begin{bmatrix} 1 & t & c1^i_k\cdot t^2/2 & 0 & 0 & c2^i_k\cdot t^2/2 & 0 & 0 & c3^i_k\cdot t^2/2 \\ 0 & 1 & c1^i_k\cdot t & 0 & 0 & c2^i_k\cdot t & 0 & 0 & c3^i_k\cdot t \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & c4^i_k\cdot t^2/2 & 1 & t & c5^i_k\cdot t^2/2 & 0 & 0 & c6^i_k\cdot t^2/2 \\ 0 & 0 & c4^i_k\cdot t & 0 & 1 & c5^i_k\cdot t & 0 & 0 & c6^i_k\cdot t \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & c7^i_k\cdot t^2/2 & 0 & t & c8^i_k\cdot t^2/2 & 1 & t & c9^i_k\cdot t^2/2 \\ 0 & 0 & c7^i_k\cdot t & 0 & 0 & c8^i_k\cdot t & 0 & 1 & c9^i_k\cdot t \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \tag{28}$$

The gravity is treated as a deterministic input. When the direction of the Z-axis of the fixed frame is chosen opposite to the local downwards, the velocity and the position changes due to gravity measurements should be compensated for along this direction. Then, the system input matrix becomes

$$\Gamma_k \cdot u_{k-1} = [\, 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad -|g_k|\cdot t^2/2 \quad -|g_k|\cdot t \quad 0 \,]^{\mathrm{T}}. \tag{29}$$

Since the gravity can be assumed constant for short distance navigation, the system input matrix can also be assumed constant.

The acceleration and the position components are directly measured using calibrated accelerometers and position sensor. Then, the measurement matrix in (6) can be written as

$$H_k = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \tag{30}$$

$$^f_bC^i_k = \begin{bmatrix} c1^i_k & c2^i_k & c3^i_k \\ c4^i_k & c5^i_k & c6^i_k \\ c7^i_k & c8^i_k & c9^i_k \end{bmatrix} = \begin{bmatrix} q0^{i^2}_k + q1^{i^2}_k - q2^{i^2}_k - q3^{i^2}_k & 2\left(q1^i_k q2^i_k - q0^i_k q3^i_k\right) & 2\left(q1^i_k q3^i_k + q0^i_k q2^i_k\right) \\ 2\left(q1^i_k q2^i_k + q0^i_k q3^i_k\right) & q0^{i^2}_k - q1^{i^2}_k + q2^{i^2}_k - q3^{i^2}_k & 2\left(q2^i_k q3^i_k - q0^i_k q1^i_k\right) \\ 2\left(q1^i_k q3^i_k - q0^i_k q2^i_k\right) & 2\left(q2^i_k q3^i_k + q0^i_k q1^i_k\right) & q0^{i^2}_k - q1^{i^2}_k - q2^{i^2}_k + q3^{i^2}_k \end{bmatrix} \tag{22}$$

$$x^i_{\mathrm{KF},k} = [\, P^i_{kx,k} \quad V^i_{kx,k} \quad A^i_{kx,k} \quad P^i_{ky,k} \quad V^i_{ky,k} \quad A^i_{ky,k} \quad P^i_{kz,k} \quad V^i_{kz,k} \quad A^i_{kz,k} \,]^{\mathrm{T}} \tag{27}$$

C. Orientation Filtering Technique

The proposed method uses a PF to estimate the orientation of an object. Since there is no deterministic input to the system for the orientation, the deterministic input is zero at every time step. The weight of each particle can be determined using the position calculation of each orientation particle and the position measurements as the expected value. In such approach, however, the differences between the position measurements and the position calculation of each orientation state can be a result of not only the orientation state but also the sensor errors such as noise. Then, the particles with higher orientation error can have higher likelihood of being the correct orientation. In other word, the position measurements at time t_k may not be accurate enough to approximate the true posterior of orientation.

In order to overcome this problem, the summation of the position difference for a period of time (ΔT_s, where subscript s is the sth orientation iteration, $s = 1, 2, \ldots$) is used to determine the weights instead of instantaneous position difference at time t_k. In addition, instead of using the direct position measurements, the KF position estimations of each particle, which incorporate the position measurements, are used to reduce the effect of the measurement noise. Then, the likelihood is calculated based on the accumulated position difference between the estimations and the calculated values of the ith particle as

$$PE_s^i = \sum_{k=(s-1)\cdot M_s+1}^{M_s \cdot s} \left\{ \left(P_{px,k}^i - P_{kx,k}^i\right)^2 + \left(P_{py,k}^i - P_{ky,k}^i\right)^2 + \left(P_{pz,k}^i - P_{kz,k}^i\right)^2 \right\} \quad (31)$$

where $M_s = \Delta T_s / t$, and $P_{p\text{-axis},k}^i$ are the position states of the ith orientation particle at time t_k. The lower PE_s^i of the given particle signifies the higher likelihood of being the correct orientation. For every ΔT_s period elapses, the weight of each particle is recalculated based on PE_s^i values. Then, the PF should be modified to use PE_s^i instead of the direct position measurements. Hence, the posterior approximation up to time t_s becomes

$$p\left(x_{0:s} \mid PE_{1:s}^i\right) \approx \sum_{i=1}^{N} w_k^i \delta\left(x_{0:s} - x_{0:s}^i\right). \quad (32)$$

Then, the posterior up to time t_s is expressed as

$$p\left(x_{0:s} \mid PE_{1:s}^i\right) \propto p\left(PE_s^i \mid x_s\right) \cdot p(x_s \mid x_{s-1}) \cdot p\left(x_{0:s-1} \mid PE_{1:s-1}^i\right) \quad (33)$$

and the normalized weight at the sth orientation iteration is

$$w_s^i \propto \frac{p\left(PE_s^i \mid x_s^i\right) \cdot p\left(x_s^i \mid x_{s-1}^i\right)}{r\left(x_s^i \mid x_{s-1}^i, PE_s^i\right)} \cdot w_{s-1}^i. \quad (34)$$

By resampling and choosing the importance density from prior $p(x_s \mid x_{s-1})$, the normalized weight has the following relationship:

$$w_s^i \propto p\left(PE_s^i \mid x_{\mathrm{PF},s}^i\right). \quad (35)$$

TABLE I
SUMMARY OF THE PROPOSED PF

1. If $k = 1$, Draw x_0^i from $p(x_0)$ and $w_0^i = 1/N$	// Initialization
2. Calculate $x_{PF,k}^i$ according to (24)	// Prediction
3. Calculate PE_s^i according to (31)	// Likelihood
4. IF remain $(k / M_s) = 0$	// Update
$x_s^i = x_{PF,k}^i$	// Predicted states
Calculate $w^{*i}_s = \exp\left(\frac{-(PE_s^i - \arg\min(PE_s^i))^2}{2\times(\sigma(PE_s^i))^2}\right)$	// Weight calculation
$w_s^i = w^{*i}_s / \sum_{i=1}^{N} w^{*i}_s$	// Normalize weights
Draw x_s^i based on w_s^i	// Resampling
$w_s^i = 1/N$	// Reset weights
$x_{PF,k}^i = x_s^i$	
End IF	

The weight of the orientation particle is calculated based on PE_s^i and the most probable value of PE_s^i. $\arg\min(PE_s^i)$ is used as the most probable value because the orientation with the minimum accumulated position error will have the highest likelihood of being the correct orientation. Then, the normalized weight is calculated as

$$w_s^i \propto \exp\left(\frac{-\left(PE_s^i - \arg\min\left(PE_s^i\right)\right)^2}{2 \times \left(\sigma\left(PE_s^i\right)\right)^2}\right) \quad (36)$$

where $\sigma(PE_s^i)$ is the standard deviation of PE_s^i. Since $3\sigma(PE_s^i)$ contains 99.7% of the values of PE_s^i, it is valid to assume that the mean of PE_s^i less argmin(PE_s^i) is $3\sigma(PE_s^i)$. After resampling, the quaternion terms must be normalized to satisfy (23). The summary of the proposed particle filtering technique is described in Table I.

Note that ΔT_s should be chosen based on the sensor accuracy. If ΔT_s is too small, the accumulated position difference may not be large enough to help identify the best orientation particle. When ΔT_s is too large, the orientation error may become large prior to the application of the orientation evaluation. This will, in turn, impact the orientation estimation accuracy. In addition, this algorithm requires more particles to represent the pdf in order to cover a wider range of possible orientation angles, which leads to higher computational cost.

D. Angular Velocity Correction Using an Expert System

Bayesian estimation techniques are suitable for estimating dynamic states, but an alternative approach is required for estimating the stationary state. In order to identify the stationary state of the object and correct angular velocity, an expert system is developed as a part of the estimation system.

An expert system is a decision-making algorithm that uses expert knowledge. The knowledge base is constructed with expert knowledge and rules. Rules are usually expressed in a form of "IF A THEN C" where A is a set of antecedent conditions and C is a set of consequences.

The expert system in the proposed approach is used in two cases: 1) estimating the initial gyro biases and 2) detecting the stationary state. Both gyros and accelerometers suffer from biases and gain errors mainly caused by temperature drift. Thus, many commercial IMUs have temperature drift compensation to reduce the effects of biases and gain errors. However, the gains and biases of an IMU are different whenever the sensor is switched on. Although it would be beneficial to calibrate the accelerometer before use, the effects of accelerometer biases and gain errors due to switch-on could be insignificant when the IMU/position sensor hybrid system is used. This is because the gain and bias differences due to switch-on are usually very small and a position sensor is used to correct the velocity and position estimation errors. However, gyro biases lead to orientation drift over time. The simulation results of the proposed method show that the orientation error increases as the biases increase for a given size of particles [21]. To maintain the accuracy with higher angular velocity biases, the proposed filter will require more particles. This will, in turn, result in added computational complexity. Therefore, the biases of gyros should be minimized before they are used for the orientation estimation.

In order to reduce the biases of gyros, the object is left stationary for 1 s, and the mean value of each axis of gyro is subtracted from the gyro measurements. The rules for the initial bias compensation can be from Rule 1 to Rule 3: where ω_sum_{axis} is the angular velocity summation of each axis, $\omega\_bias_{\text{axis}}$ is the calculated angular velocity bias of each axis, and $\omega\_meas_{\text{axis}}$ is the angular velocity measurement of each axis.

Rule 1: IF time ≤ 1 s THEN
$\omega\_sum_{\text{axis}} = \omega\_sum_{\text{axis}} + \omega\_meas_{\text{axis}}$ AND $\omega_{\text{axis}} = 0$
Rule 2: IF time $= 1$ s THEN
$\omega\_bias_{\text{axis}} = \omega\_sum_{\text{axis}} \times t$
Rule 3: IF time > 1 s THEN
$\omega_{\text{axis}} = \omega\_meas_{\text{axis}} - \omega\_bias_{\text{axis}}$

Another role of the expert system is to identify the stationary state. When the object is identified stationary, the angular velocity is changed to zero. Even when the angular velocity is zero, the yaw angle can still drift during the resampling step. This is due to the fact that the yaw angle drift does not affect the position calculation when the object is stationary. Therefore, while the object is stationary, the resampling step is suspended to ensure the current orientation state is the same as the previous orientation state ($x^i_{\text{PF},k} = x^i_{\text{PF},k-1}$). In order to suspend the resampling step, ΔT_s is increased by one sample time whenever the object is identified stationary ($\Delta T_s = \Delta T_s + t$).

The steady state is identified using the corrected angular velocities, acceleration measurements, and position measurements. When the object is stationary, the angular velocity of the object is zero and the acceleration and position of each axis are constants. From a measurement stand point, the magnitude of the angular velocity of each axis should be lower than the maximum angular velocity error of the gyro when the object is stationary [31]. In addition, the acceleration fluctuation is smaller than the maximum acceleration noise from the accelerometer. In addition, the fluctuation of the position measurement is less than the maximum error of the position sensor. The acceleration fluctuation (Acc_fluc_{axis}) and the position measurement fluctuation (Pos_fluc_{axis}) of each axis are defined as

$$Acc\_fluc_{\text{axis}} = |a_{\text{axis}} - Avg\_A_{\text{axis}}| \tag{37}$$

$$Pos\_fluc_{\text{axis}} = |p_{\text{axis}} - Avg\_P_{\text{axis}}| \tag{38}$$

where a_{axis} is the acceleration measurement of each axis, p_{axis} is the position measurement of each axis, and the average acceleration of each axis (Avg_A_{axis}) and the average position of each axis (Avg_p_{axis}) are expressed as

$$Avg\_A_{\text{axis}} = \sum_{j=1}^{n} a_{\text{axis}}(j)/n \tag{39}$$

$$Avg\_P_{\text{axis}} = \sum_{j=1}^{n} p_{\text{axis}}(j)/n \tag{40}$$

where n is the number of time steps while the object is stationary. When the object moves, n is reset to unity.

The expert system in Fig. 1 incorporates Rule 4 to remove the orientation drift while the object is stationary: where $\omega_{\text{axis\_max\_error}}$ is the maximum angular velocity error in each axis, $A_{\text{axis\_max\_noise}}$ is the maximum acceleration error in each axis, and $P_{\text{axis\_max\_error}}$ is the maximum position error in each axis. The $t_{\text{stationary}}$ period should be chosen depending on the application.

Rule 4:
IF $|\omega_{\text{axis}}| \leq \omega_{\text{axis\_max\_error}}$ for the last $t_{\text{stationary}}$ seconds AND $Acc\_fluc_{\text{axis}} < a_{\text{axis\_max\_noise}}$ for the last $t_{\text{stationary}}$ seconds AND $Pos\_fluc_{\text{axis}} < P_{\text{axis\_max\_error}}$ for the last $t_{\text{stationary}}$ seconds,
THEN $\omega_{\text{axis}} = 0$, and $\Delta T_s = \Delta T_s + t$

E. Initial Orientation Estimation

In order to use PF, the initial pdf of the orientation state needs to be identified. Two extreme cases can be considered: 1) when the orientation state is almost completely known and 2) when the orientation of the object is completely unknown. When the initial orientation of the object is known with a high level of certainty, the initial pdf can be represented as a normal distribution with a very small covariance.

However, in many circumstances, the initial orientation is completely unknown. Then, two different approaches are possible. First, quaternion states are drawn from a uniform distribution. In this case, a high initial process noise covariance is selected because the uncertainty of the orientation state is initially high. It is assumed that the orientation converges to the correct value as time elapses. Then, the uncertainty also reduces over time. In order to incorporate this knowledge, the process

noise covariance monotonically reduces to the final value where the orientation is known with a high level of certainty.

The second approach is to utilize the knowledge that the object is stationary for the first 1 s. Then, the average acceleration measurements for the first 1 s can be used to calculate the tilt angles [31] so that the uncertainties in roll and pitch angles (θ_x and θ_y, respectively) are reduced. The relationship between roll and pitch angles and the acceleration measurements due to gravity is given by

$$\theta_x = \arctan(Avg\_A_{y\_i}/Avg\_A_{z\_i})$$
$$\theta_y = \arcsin\left(\frac{-Avg\_A_{x\_i}}{\left(Avg\_A_{x\_i}^2 + Avg\_A_{y\_i}^2 + Avg\_A_{z\_i}^2\right)^{0.5}}\right) \tag{41}$$

where $Avg\_A_{\text{axis}\_i}$ is the initial average acceleration of each axis for the first 1 s using (39).

Although roll and pitch angles can be determined from (41), the uncertainty in yaw angle remains an issue. If any information about the yaw angle is available, it can be used to construct the initial pdf. However, if no information is available about the initial state of the yaw angle, a uniform distribution can be used to represent the pdf of the yaw angle. If 20 particles are used, the particles are placed 18° (360°/20 particles) apart in the yaw angle domain. Then, the initial quaternion states using these angles are defined as [29]

$$\begin{aligned}
q0_0^i &= \cos\left(\frac{\theta_x}{2}\right)\cdot\cos\left(\frac{\theta_y}{2}\right)\cdot\cos\left(\frac{\theta_z^i}{2}\right) \\
&\quad + \sin\left(\frac{\theta_x}{2}\right)\cdot\sin\left(\frac{\theta_y}{2}\right)\cdot\sin\left(\frac{\theta_z^i}{2}\right) \\
q1_0^i &= \sin\left(\frac{\theta_x}{2}\right)\cdot\cos\left(\frac{\theta_y}{2}\right)\cdot\cos\left(\frac{\theta_z^i}{2}\right) \\
&\quad - \cos\left(\frac{\theta_x}{2}\right)\cdot\sin\left(\frac{\theta_y}{2}\right)\cdot\sin\left(\frac{\theta_z^i}{2}\right) \\
q2_0^i &= \cos\left(\frac{\theta_x}{2}\right)\cdot\sin\left(\frac{\theta_y}{2}\right)\cdot\cos\left(\frac{\theta_z^i}{2}\right) \\
&\quad + \sin\left(\frac{\theta_x}{2}\right)\cdot\cos\left(\frac{\theta_y}{2}\right)\cdot\sin\left(\frac{\theta_z^i}{2}\right) \\
q3_0^i &= \cos\left(\frac{\theta_x}{2}\right)\cdot\cos\left(\frac{\theta_y}{2}\right)\cdot\sin\left(\frac{\theta_z^i}{2}\right) \\
&\quad + \sin\left(\frac{\theta_x}{2}\right)\cdot\sin\left(\frac{\theta_y}{2}\right)\cdot\cos\left(\frac{\theta_z^i}{2}\right)
\end{aligned} \tag{42}$$

where θ_z^i is the ith yaw angle at 1 s.

Similar to the first approach, high initial process noise covariance is selected and monotonically reduced until it reaches the final value. However, since this algorithm already has the roll and pitch angle information, the initial process noise covariance should be selected lower than the level chosen for the first approach of case 2).

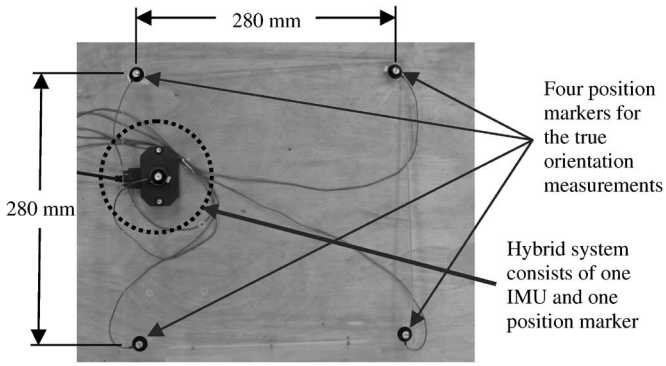


Fig. 2. Proposed hybrid system and the true orientation measurement system.

IV. Experiments

The proposed method was experimentally tested using a hybrid system that consists of one IMU (3DM-GX2, Microstrain) and one position sensor (Optotrak, NDI). Optotrak can locate infrared light-emitting diode (IRED) position markers within submillimeter error. One IRED position marker was attached to the center of the IMU. To check the orientation accuracy of the proposed method, true orientation was calculated from four IRED position markers placed 280 mm apart from each other. The configuration of the sensors is shown in Fig. 2. Optotrak requires lines of sight between the IRED position markers and the camera. Since an IRED position marker has limited signal-emitting angle, three Optotrak cameras were used to track the position of the markers.

In this experiment, the sensors were moved by hand in random 3-D motions for 8 min and returned to the original position and orientation. The sensor was kept stationary for the first 1 s to remove the gyro bias as well as for the last 10 s to check if the expert system can detect the stationary state. The true position and orientation measurements using Optotrak are shown in Fig. 3.

Both position sensor and the IMU were sampled at 100 Hz ($t = 0.01$ s). The proposed method evaluates the weights of the particles every 1-s interval ($\Delta T_{1:s} = 1$ s). Since it is unlikely to move an object and satisfy Rule 4 in Section III for 1 s when the object is moved by a person, $t_{\text{stationary}}$ is set to 1 s. Four different analyses were conducted for the same test: 1) orientation error comparison between the EKF and the proposed filter; 2) orientation error comparison of the proposed filters with different numbers of particles; 3) orientation error comparison of the proposed filters with different numbers of particles when the initial condition is completely unknown, and 4) position and orientation comparison when Gaussian noise is added to the position measurements.

First, the orientation errors obtained with both an EKF and the proposed filter with 20 particles are compared in Fig. 4. The expert system described in Section III was applied to both EKF and the proposed filter to correct the angular velocity. Fig. 4 shows that the angle errors using the EKF increase over time, but the errors using the proposed filter are significantly reduced and do not grow over time. The roll and pitch angle errors of the proposed method have lower magnitudes than the yaw angle error because the roll and pitch angles are associated with the

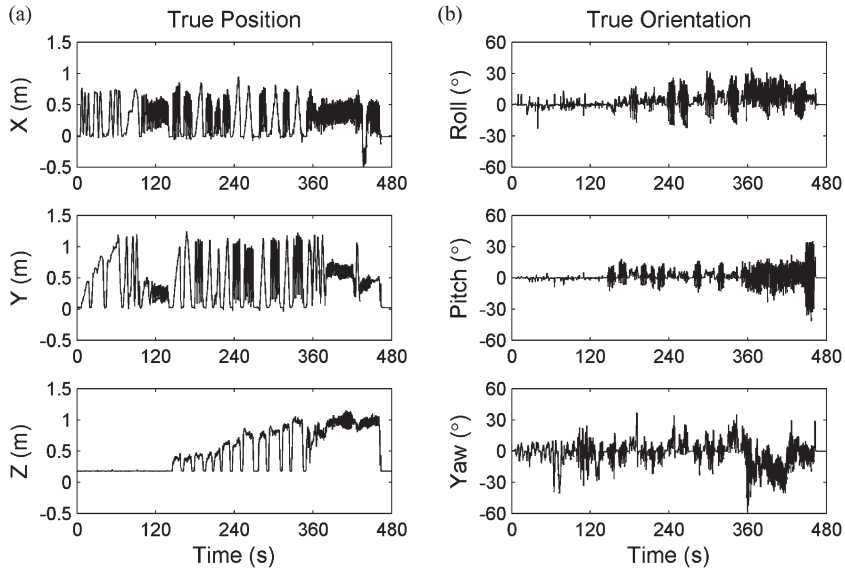


Fig. 3. True (a) position and (b) orientation measurements using Optotrak.

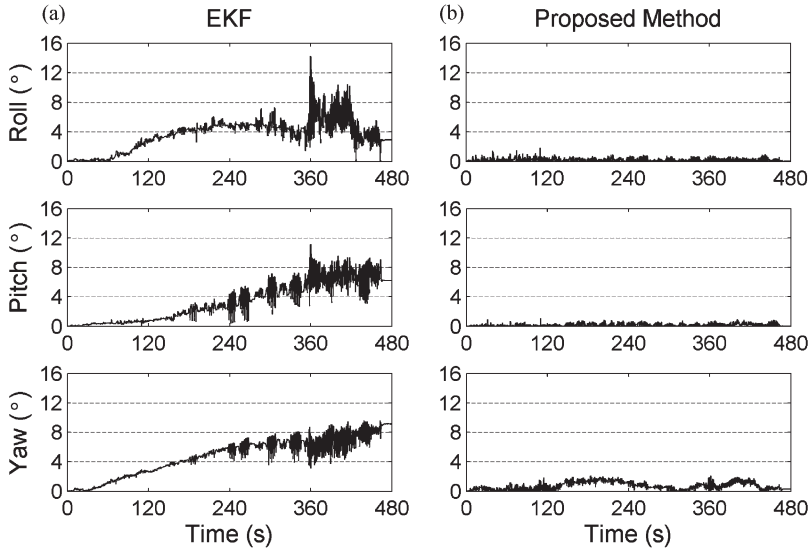


Fig. 4. Orientation errors using (a) EKF and (b) the proposed method with 20 particles.

gravity vector when the acceleration of the object is calculated. As a result, the orientation particles with higher roll and pitch angle errors have lower weights and die out faster. To analyze the orientation estimation accuracy better, the root-mean-square (rms) errors of the rotation matrices are compared in Fig. 5. The results show that the rotation matrix error using an EKF increases over time. When the proposed method was applied, the rotation matrix error is significantly reduced and the error does not increase over time. The errors of the last 10 s in Figs. 4 and 5 did not change because the angular velocity is reduced to zero when the object is stationary.

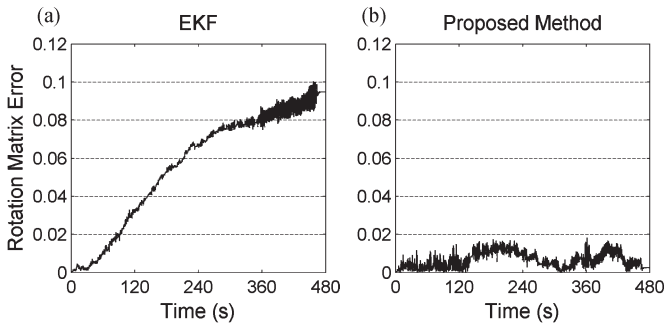


Fig. 5. RMS rotation matrix error using (a) EKF and (b) the proposed method with 20 particles.

For the second analysis, the orientation error was calculated using the proposed method with 5, 20, and 80 particles. Figs. 6 and 7 show the corresponding orientation errors in each case. When the number of particles is increased from 5 to 20, the orientation error is significantly reduced. However, when the number of particle is increased from 20 to 80, the graphs show no significant improvement. This experiment shows that increasing the number of particles does not proportionally improve orientation estimation accuracy.

For the third analysis, the initial orientation was assumed completely unknown, and the orientation of the object was

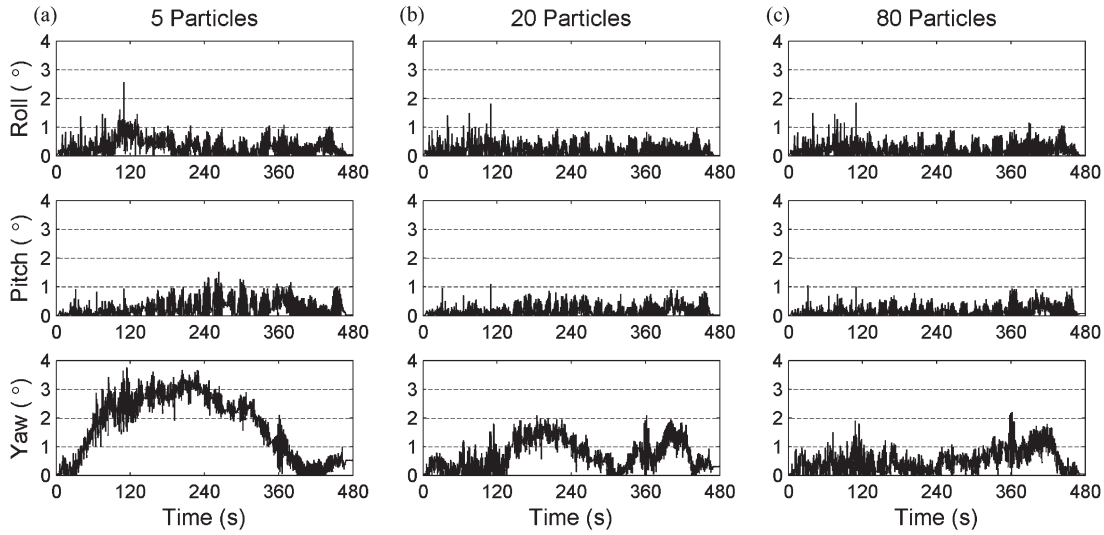


Fig. 6. Euler angle errors using the proposed method when the initial orientation is known. (a) 5 particles. (b) 20 particles. (c) 80 particles.

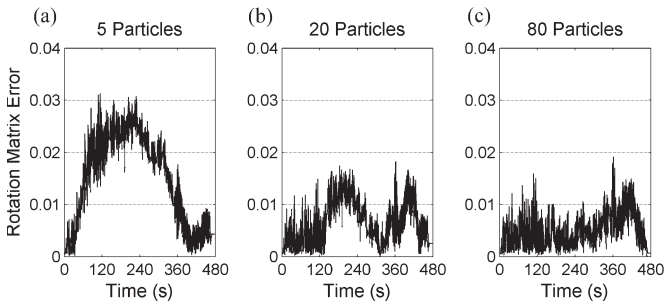


Fig. 7. RMS rotation matrix errors using the proposed method when the initial orientation is known. (a) 5 particles. (b) 20 particles. (c) 80 particles.

estimated using the proposed filters with 5, 20, and 80 particles. The initial orientation is distributed using (42) at 1 s. In addition, ΔT_1 is chosen 3 s instead of 1 s so that the majority of particles with small weights die out in the first orientation iteration. When the initial orientation is unknown, high initial process noise is selected and then decreased gradually over time to its final value used in the previous two analyses.

The initial process noise and the settling time are reduced as the number of particles increase. When five particles are used, the initial process noise is 40 times higher than the final value. Then, the process noise converges monotonically to the final value in 390 s. When 20 particles are used, the initial process noise is 16 times higher than its final value and converges to this value in 150 s. When 80 particles are used, the initial process noise is four times higher than its final value. This process noise converges to the final value in 30 s.

Fig. 8 shows the absolute values of Euler angle errors (the actual roll and pitch angle errors vary from -90° to 90°, and the yaw angle error varies from -180° to 180°) for the aforementioned three scenarios when the initial orientation is completely unknown. Fig. 8(a) shows the orientation error with the proposed filter with five particles. Since only five particles are used to span the entire yaw angle range, this approximation cannot represent the true pdf very well. As a result, the particle with high yaw angle error (120°) has the highest weight at the beginning and slowly converges to the correct orientation. Fig. 8(a) shows that the orientation converges to the correct value as time elapses even though the initial value was far from the correct orientation. When the yaw angle is around 90°, the roll and pitch angle errors become unstable in Fig. 8(a), which is a well-known Euler angle problem. The last row of Fig. 8 shows the magnified yaw angle errors. The plots in the last rows of Fig. 8 show that as time elapses, the yaw angle errors almost match their counterparts in Fig. 6. The last row of Fig. 8(b), the proposed method with 20 particles, shows that the correct orientation was found at 70 s. However, the yaw angle error started increasing significantly because the number of particles is small and the process noise is high. Since the process noise is designed to reduce monotonically to the final value in 150 s, the yaw angle error does not significantly increase after 120 s. Fig. 8 shows that the proposed filter finds the correct orientation faster as the number of particles increase.

For a close-up view of Fig. 8(c), the first 15 s of the proposed method with 80 particles are shown in Fig. 9. Initially, the orientation particles were randomly drawn from a uniform distribution. At 1 s, the roll and pitch angels were estimated using (41). At this instance, the roll and pitch angle errors are close to zero, and the yaw angle is represented by 80 particles that are uniformly distributed. Around 6 s, after 3 s of movement period excluding the stationary state, the orientation particle with about 130° difference from the correct yaw angle is chosen as output because it has the highest weight. At 7 s, however, the yaw angle error is considerably reduced indicating that the filter contains a set of particles with low orientation errors. In other words, since 80 particles are used, many particles with low yaw angle error survived during resampling at 6 s and chosen as output at 7 s.

The same test was repeated with 20 particles without using (41) to estimate the initial roll and pitch angles. The initial process noise is 26 times higher than the final value and is allowed to converge to the final value in 250 s. Fig. 10 shows the experimental results. As expected, this method requires a longer time to converge to the correct orientations due to the uncertainties in roll and pitch angles.

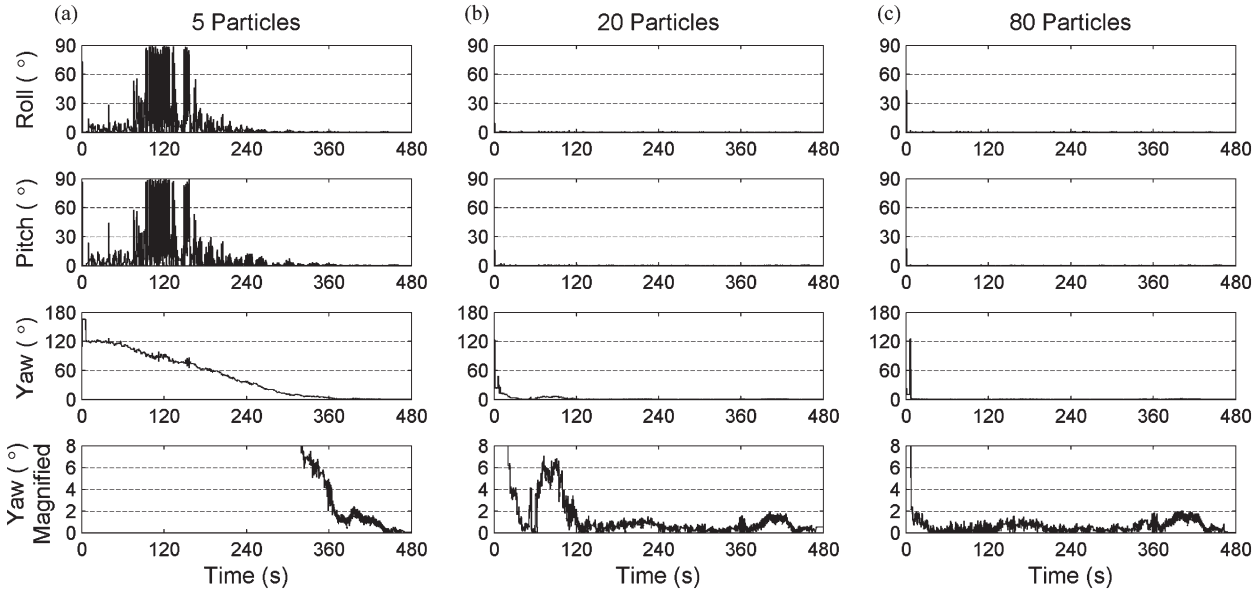


Fig. 8. Euler angle errors when the initial orientation is unknown: proposed method with (a) 5 particles, (b) 20 particles, and (c) 80 particles.

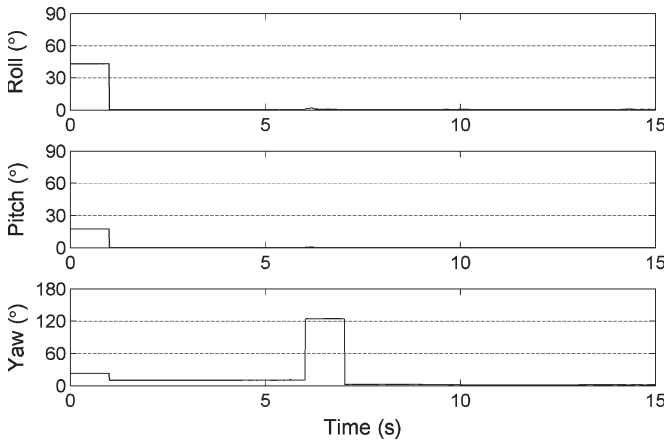


Fig. 9. First 15 s of Fig. 8(c), the orientation error with 80 particles when the initial orientation is unknown.

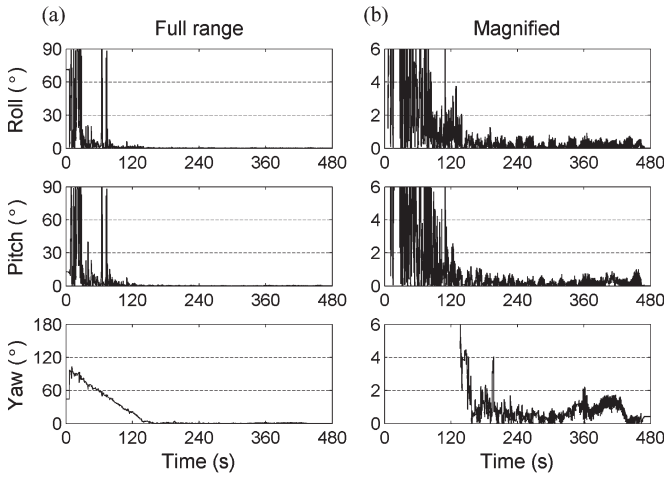


Fig. 10. Euler angle errors using the proposed method with 20 particles when the initial orientation is unknown and accelerometer measurements are not used to estimate roll and pitch angles. (a) Full range. (b) Range from 0° to 6°.

Lastly, the effect of position measurement noise on the orientation error is studied. To investigate the effects of the position measurement noise on the orientation and position accuracies, the same experiment was repeated with added Gaussian noise to the Optotrak position measurements. In this analysis, the position measurements are treated as the true position because Optotrak has only submillimeter error. Fig. 11 shows the orientation error using the proposed method with added position noise, and Fig. 12 shows the corresponding position error. The plots in Fig. 11 show lower orientation errors than those obtained using the EKF in Fig. 4(a). This is particularly evident for the roll and pitch angles. This result shows that the proposed method has a good performance even in the presence of additional Gaussian position noise. As expected, Figs. 6 and 11 show that the position measurement noise tends to reduce the orientation accuracy. Fig. 11 also shows that the orientation accuracy increases as the number of particle increases. However, increasing the number of particles increases the computational cost, as shown in Table II. The proposed method was calculated using Intel Core2 Duo Processor E8400 with 3-Gb memory. Matlab was used to analyze the data offline. Based on the results in Table II, the proposed method with 80 particles is not feasible for online application with the current system because the processing time exceeds the experiment duration, 480 s. However, the processing time can be reduced by using faster processor or more dedicated solvers such as those that utilize C or Java development platform. It can also be concluded from the results in Table II that the number of particles should be chosen to achieve a tradeoff between computational power and estimation accuracy.

Finally, Fig. 12 shows the position rms error using three different approaches for position estimation: 1) no filter (added position noise); 2) an EKF; and 3) the proposed methods with three different particle sizes. The position estimations of both EKF and the proposed methods show improved results over the case when no filter was used. The proposed methods in all

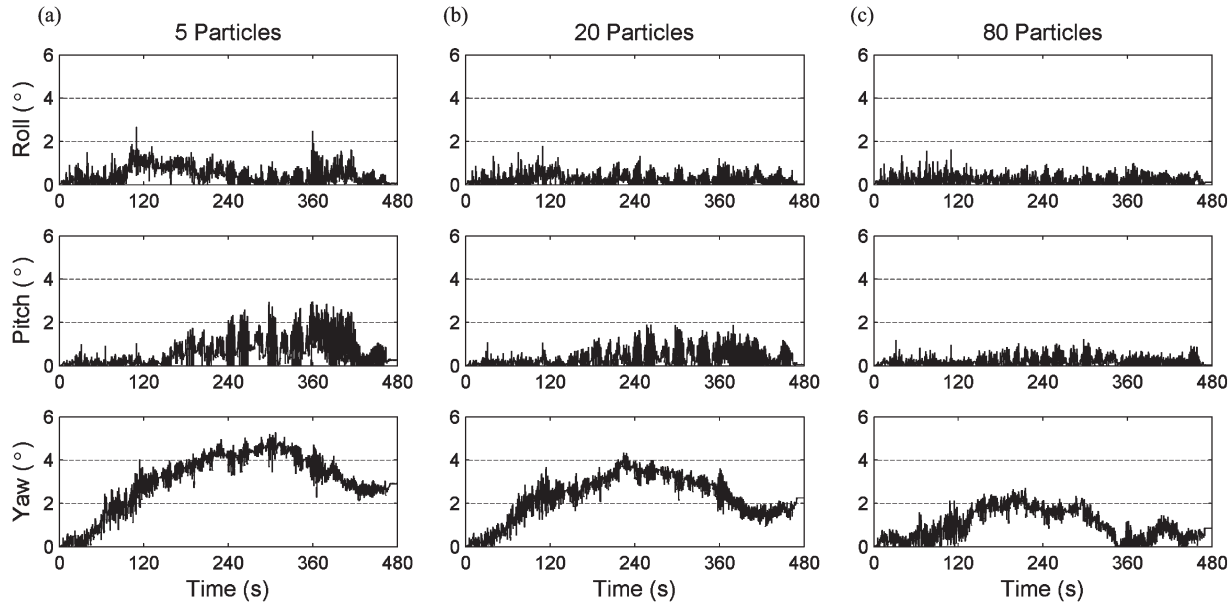


Fig. 11. Euler angle error with the proposed method when Gaussian noise is added to the position measurement (a) with 5 particles, (b) with 20 particles, and (c) with 80 particles.

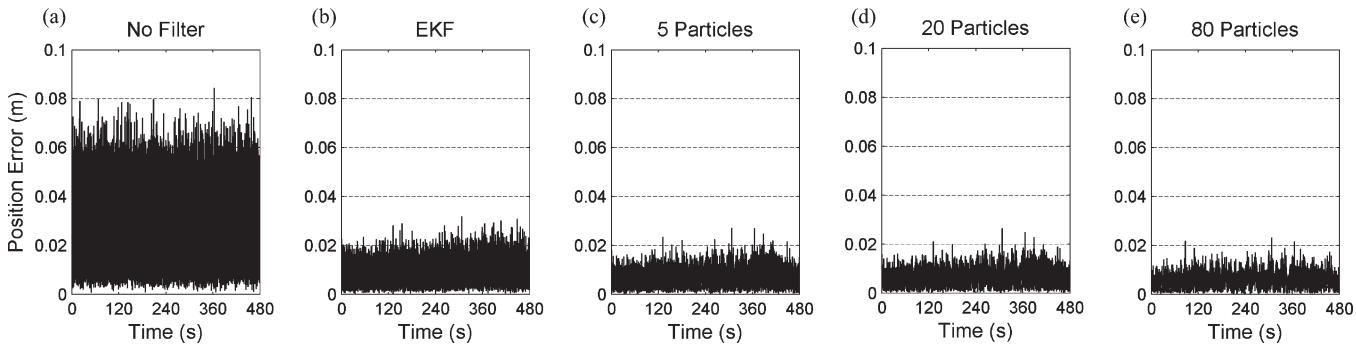


Fig. 12. Position error when Gaussian noise is added to the position measurements. (a) Added Gaussian noise using no filter. (b) Using an EKF. (c) Using the proposed filter with five particles. (d) Using the proposed filter with 20 particles. (e) Using the proposed filter with 80 particles.

TABLE II
NUMBER OF PARTICLES AND THEIR PROCESSING TIME

Number of particles	5	20	80
Processing time	43 s	142 s	521 s

three cases yield lower position error than the EKF estimation method. Fig. 12 also suggests that the proposed method provides higher position accuracy when the number of particles increases.

V. CONCLUSION

This paper presented a position/orientation estimation method using a KF/PF combination. The proposed method uses PF to estimate object orientation while KF is used to estimate the position. In addition, an expert system was used to correct the angular velocity bias as well as to identify the stationary state of the object.

Experimental results show that the orientation errors using the proposed method are greatly reduced compared to errors obtained using only the EKF estimation method. Two factors that affect the orientation error were studied: 1) number of particles and 2) position sensor noise. As the number of particles increases, the orientation error using the proposed method tends to decrease. However, increasing the number of particles requires higher computational costs; hence, the number of particles should be chosen to achieve a tradeoff between accuracy and computational efficiency. The experimental results show that the proposed method can estimate orientation even in the presence of Gaussian position sensor noise. However, the experimental results show that position noise decreases the orientation estimation accuracy. The experimental results also show that the proposed filter can find the correct orientation of the object even when the initial orientation is completely unknown.

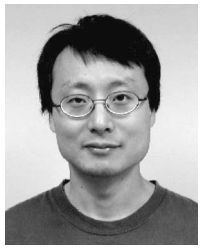
REFERENCES

[1] X. Yun, E. R. Bachmann, R. B. McGhee, R. H. Whalen, R. L. Roberts, R. H. Knapp, A. J. Healey, and M. J. Zyda, "Testing and evaluation of an integrated GPS/INS system for small AUV navigation," *IEEE J. Ocean. Eng.*, vol. 24, no. 3, pp. 396–404, Jul. 1999.

[2] K. Kobayashi, K. C. Cheok, K. Watanabe, and F. Munekata, "Accurate differential global positioning system via fuzzy logic Kalman filter sensor fusion technique," *IEEE Trans. Ind. Electron.*, vol. 45, no. 3, pp. 510–518, Jun. 1998.

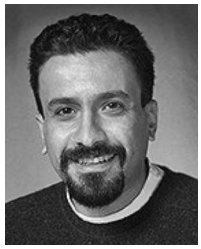
[3] G. Hyslop, D. Gerth, and J. Kraemer, "GPS/INS integration on the stand-off land attack missile (SLAM)," *IEEE Aerosp. Electron. Syst. Mag.*, vol. 5, no. 7, pp. 29–34, Jul. 1990.

[4] N. Parnian and M. F. Golnaraghi, "Integration of vision and inertial sensors for industrial tools tracking," *Sens. Rev.*, vol. 27, no. 2, pp. 132–141, 2007.
[5] R. Oktem, E. Aydin, and N. Cagiltay, "An indoor navigation aid designed for visually impaired people," in *Proc. 34th IEEE IECON*, Orlando, FL, Nov. 10–13, 2008, pp. 2982–2987.
[6] J. Yang, E.-S. Choi, W. Chang, W.-C. Bang, S.-J. Cho, J.-K. Oh, J.-K. Cho, and D.-Y. Kim, "A novel hand gesture input device based on inertial sensing technique," in *Proc. IEEE IECON*, Busan, Korea, Nov. 2–6, 2004, pp. 2786–2791.
[7] Y. Tao, H. Hu, and H. Zhou, "Integration of vision and inertial sensors for 3D ARM motion tracking in home-based rehabilitation," *Int. J. Robot. Res.*, vol. 26, no. 6, pp. 607–624, Jun. 2007.
[8] S. P. Won, W. W. Melek, and F. Golnaraghi, "A fastening tool tracking system using an IMU and a position sensor with Kalman filters and a fuzzy expert system," *IEEE Trans. Ind. Electron.*, vol. 56, no. 5, pp. 1782–1792, May 2009.
[9] X. He, X. Hu, W. Wu, and M. Wu, "Intelligent SINS/RDSS integrated algorithms for land vehicle navigation," *IEEE Aerosp. Electron. Syst. Mag.*, vol. 24, no. 3, pp. 4–11, Mar. 2009.
[10] K. Touil, M. Zribi, J. B. Choquel, and M. Benjelloun, "Bayesian bootstrap filter for integrated GPS and dead reckoning positioning," in *Proc. IEEE ISIE*, Vigo, Spain, 2007, pp. 1520–1524.
[11] J. L. Crassidis, "Sigma-point Kalman filtering for integrated GPS and inertial navigation," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 42, no. 2, pp. 750–756, Apr. 2006.
[12] E. Foxlin and L. Naimark, "VIS-Tracker: A wearable vision-inertial self-tracker," in *Proc. IEEE Virtual Reality*, Los Angeles, CA, Mar. 22–26, 2003, pp. 199–206.
[13] F. Gustafsson, F. Gunnarsson, N. Bergman, U. Forssell, J. Jansson, R. Karlsson, and P.-J. Nordlund, "Particle filters for positioning, navigation, and tracking," *IEEE Trans. Signal Process.*, vol. 50, no. 2, pp. 425–437, Feb. 2002.
[14] N. Yang, W. F. Tian, Z. H. Jin, and C. B. Zhang, "Particle filter for sensor fusion in a land vehicle navigation system," *Meas. Sci. Technol.*, vol. 16, no. 3, pp. 677–681, Mar. 2004.
[15] P.-J. Nordlund and F. Gustafsson, "Sequential Monte Carlo filtering techniques applied to integrated navigation systems," in *Proc. Amer. Control Conf.*, 2001, vol. 6, pp. 4375–4380.
[16] D. Lin, L. Keck Voon, H. Guo Rong, and N. Nagarajan, "GPS-based attitude determination for microsatellite using three-antenna technology," in *Proc. Aerosp. Conf.*, Mar. 6–13, 2004, vol. 2, pp. 1024–1029.
[17] G. Lu, M. E. Cannon, and G. Lachapelle, "Attitude determination in a survey launch using multi-antenna GPS technologies," in *Proc. ION Nat. Tech. Meeting*, San Francisco, CA, Jan. 20–22, 1993, pp. 251–260.
[18] E. Cattrysse, S. Provyn, P. Kool, O. Gagey, J. Clarys, and P. Van Roy, "Reproducibility of kinematic motion coupling parameters during manual upper cervical axial rotation mobilization: A 3-dimensional in vitro study of the atlanto-axial joint," *J. Electromyogr. Kinesiol.*, vol. 19, no. 1, pp. 93–104, Feb. 2009.
[19] R. Wolf, G. W. Hein, B. Eissfeller, and Loehnert, "An integrated low cost GPS/INS attitude determination and position location system," in *Proc. ION GPS*, Kansas City, MO, 1996, pp. 975–981.
[20] E. Foxlin, M. Harrington, and Y. Altshuler, "Miniature 6-DOF inertial system for tracking HMDs," in *Proc. SPIE—AeroSense Conf. Helmet-Head-Mounted Displays III*, Orlando, FL, 1998, vol. 3362, pp. 214–228.
[21] S. P. Won, W. W. Melek, and F. Golnaraghi, "Position and orientation estimation with an IMU and a position sensor using a Kalman filter and a particle filter," in *Proc. IEEE IECON*, Orlando, FL, Nov. 9–13, 2008, pp. 3006–3010.
[22] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*. Cambridge, MA: MIT Press, 2005.
[23] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Trans. Signal Process.*, vol. 50, no. 2, pp. 174–188, Feb. 2002.
[24] C. Yuan and M. J. Druzdzel, "Theoretical analysis and practical insights on importance sampling in Bayesian networks," *Int. J. Approx. Reason.*, vol. 46, no. 2, pp. 320–333, Oct. 2007.
[25] B.-T. Zhang, "A Bayesian framework for evolutionary computation," in *Proc. Congr. Evol. Comput.*, Washington, DC, Jul. 6–9, 1999, pp. 722–728.
[26] S. Park, J. Hwang, K. Rou, and E. Kim, "A new particle filter inspired by biological evolution: Genetic filter," *Proc. World Acad. Sci., Eng. Technol.*, vol. 21, pp. 459–463, Jan. 2007.
[27] R. Fung and K. C. Chang, "Weighting and integrating evidence for stochastic simulation in Bayesian networks," in *Proc. 5th Conf. Uncertainty Artif. Intell.*, Windsor, ON, Canada, Aug. 18–20, 1989, pp. 209–219.
[28] K. Kanazawa, D. Koller, and S. J. Russell, "Stochastic simulation algorithms for dynamic probabilistic networks," in *Proc. 11th Annu. Conf. Uncertainty AI*, 1995, pp. 346–351.
[29] D. H. Titterton and J. L. Weston, *Strapdown Inertial Navigation Technology*. London, U.K.: Peregrinus, 1997.
[30] J. C. K. Chou, "Quaternion kinematic and dynamic differential equations," *IEEE Trans. Robot. Autom.*, vol. 8, no. 1, pp. 53–64, Feb. 1992.
[31] S. Won, N. Parnian, F. Golnaraghi, and W. W. Melek, "A quaternion-based tilt angle correction method for a hand-held device using an inertial measurement unit," in *Proc. IEEE Int. Conf. Ind. Electron.*, Nov. 9–13, 2008, pp. 2971–2975.



Seong-hoon Peter Won (S'09) received the B.Sc. and M.A.Sc. degrees in mechanical engineering from the University of Waterloo, Waterloo, ON, Canada, in 2001 and 2003, respectively, where he is currently working toward the Ph.D. degree.

From 2006 to 2007, he was with the University of Waterloo, where he was involved in applications of hand-held inertial navigation systems. His main research area is improving the performance of inertial navigation systems and hybridizing an inertial measurement unit with a position sensor.



Wael William Melek (M'02–SM'06) received the M.A.Sc. and Ph.D. degrees in mechanical engineering from the University of Toronto, Toronto, ON, Canada, in 1998 and 2002, respectively.

Between 2002 and 2004, he was an AI Division Manager with Alpha Global IT, Inc., Toronto. He is currently an Assistant Professor in the Department of Mechanical Engineering, University of Waterloo, Waterloo, ON. His current research interests include mechatronics applications, robotics, industrial automation and the application of fuzzy-logic, neural networks, and genetic algorithms for modeling and control of dynamic systems.

Dr. Melek is a member of the American Society of Mechanical Engineers.

Farid Golnaraghi received the B.S. and M.S. degrees in mechanical engineering from Worcester Polytechnic Institute, Worcester, MA, in 1982, and the Ph.D. degree in theoretical and applied mechanics from Cornell University, Ithaca, NY, in 1988.

In 1988, he became a Professor of mechanical engineering, and later mechanical and mechatronics engineering, at the University of Waterloo, Waterloo, ON, Canada. He also became a Tier I Canada Research Chair in mechatronics and smart material systems at Waterloo in 2002. He joined Simon Fraser University (SFU), Burnaby, BC, Canada, in August 2006, as the Director of the Mechatronics Engineering Program, the flagship program at SFU Surrey. He is also the holder of a Burnaby Mountain Endowed Chair. Over the past 20 years at Waterloo and SFU, he has been very active in supervision of graduate students. His pioneering research has resulted in two textbooks, more than 100 papers, four patents, and two start-up companies.