DCM-Based Orientation Estimation Using Cascade of Two Adaptive Extended Kalman Filters

Anh-Tung Dang, Vinh-Hao Nguyen

Abstract – **In this paper, we propose an effective algorithm to estimate orientation angles (roll, pitch, and yaw) from Inertial Measurement Unit (IMU). This algorithm uses two adaptive extended Kalman filters (AEKF) to estimate the Direction Cosine Matrix (DCM), the external acceleration and the magnetic disturbance. First 6-state filter estimates three elements in the third row of the DCM and the external acceleration on three axes. The second one estimates three elements in the first row of the DCM and the magnetic disturbance on three axes. The last three elements of the DCM are computed from the DCM orthogonalization property. This method overcomes original problems when IMU is moved by external acceleration, and it is disturbed in magnetic environment. In addition, it helps reduce effort on computation from 15-states filter to two 6-states filters. The performance of proposed algorithm is verified by applying it to a 9-DOF IMU and testing its accuracy in various conditions. Experiment results have shown that the proposed algorithm achieves accurate estimation of orientation. The RMS error of all three angles is less than 2º.**

I. Introduction

IMU can be used to estimate orientation of a rigid body relative to an Earth fixed reference frame. To describe the orientation of a rigid body, we use Euler angles roll, pitch, and yaw relative to the rotation angles between a body and the Earth frame. IMU has a wide range of applications in robotics, virtual reality, and human motion tracking. It contains three kinds of sensor: accelerometer, gyroscope, and magnetometer. The accelerometer measures the acceleration on three axes of the IMU so that the accelerometer signals are the sum of gravitational acceleration and the external acceleration. Due to this point, we can only calculate roll and pitch angles from the accelerometer signals in static condition. The gyroscope signals provide the angular velocity on each axis of the IMU so that we can calculate all three "relative" Euler angles by integration over time step. This may lead to a bigger error on output angles over time step. The magnetometer measures magnetic field on the Earth so that it can provide a heading of the IMU or the yaw angle. However, the yaw angle may be wrong when IMU goes into disturbed magnetic environment. The authors in [2] used the Gauss-Newton method to estimate angles and applied it to human motion tracking field.

A. T. Dang is with Faculty of Electrical & Electronics Engineering, HoChiMinh City University of Technology, Vietnam. (e-mail: anhtunghcmut08@gmail.com).

V. H. Nguyen is with Faculty of Electrical & Electronics Engineering, HoChiMinh City University of Technology, Vietnam. (e-mail: vinhhao@hcmut.edu.vn).

This filter is capable of tracking a rigid body through all orientations and is more efficient than those based on Euler angles. Also, they introduced the quaternion-based Kalman filter [3], [4]. These papers present the real-time implementation and testing results of the quaternion-based Kalman filter. Experimental results validate the Kalman filter design, and show the feasibility of the MARG sensors for real-time human body motion tracking. Also in human motion tracking applications, the author in [5] developed a filter that is called complementary Kalman filter. The orientation estimated by the filter was compared with the orientation obtained with an optical reference system Vicon. Results show accurate and drift-free orientation estimates. The compensation results in a significant difference between the orientation estimates with compensation of magnetic disturbances in comparison to no compensation or only gyroscopes. The average static error was 1.4º (standard deviation 0.4) in the magnetically disturbed experiments. The dynamic error was 2.6º root means square. Another approach tries to separate the impact of acceleration and the disturbance on three angles, the author in [6] used the DCM method and Kalman filter. It has shown that normal Kalman Filter in DCM method is better than extended Kalman Filter in Euler and Quaternion based method because it helps avoid the first order approximation error. To reduce the computation effort, the author in [7] proposed gradient descent method without using Kalman filter. The author showed experimental results that his proposed algorithm achieves levels of accuracy exceeding that of the Kalman-based algorithm; <0.6º static RMS error, <0.8º dynamic RMS error.

In this paper, we introduce a fusion algorithm to estimate three Euler angles from IMU. This algorithm uses the DCM-based method that is the same approach as [6]. However, we remove angular velocity component in state variable, and directly put the gyroscope signals in the state equation. This significantly decreases the computational load for Kalman filter and increases the capable of response of the algorithm in dynamic conditions. Moreover, we expand the state equation by adding state variables including the external acceleration and the magnetic disturbance. This ensures the algorithm's accuracy during various conditions such as high acceleration, disturbed magnetic environment.

This paper is organized as follows. Section II introduces the method of developing the algorithm. Section III presents the implementation of the algorithm. Section IV describes the experiment process to validate the proposed algorithm and testing results, followed by conclusions in section V.

II. DCM METHOD & SENSOR MODEL

A. DCM

The rotation matrix ${}^{E}_{B}R$ transforms a 3x1 vector ${}^{B}v$ in sensor body frame (B) to a 3x1 vector ${}^{E}v$ in earth frame (E) as below

$$ {}^{B}v = {}^{E}_{B}R\,{}^{E}v \tag{1} $$

where ${}^{E}_{B}R$ is denoted by:

$$ {}^{E}_{B}R = \begin{bmatrix} {}^{E}_{B}R_{1i} \\ {}^{E}_{B}R_{2i} \\ {}^{E}_{B}R_{3i} \end{bmatrix} = \begin{bmatrix} {}^{E}_{B}R_{11} & {}^{E}_{B}R_{12} & {}^{E}_{B}R_{13} \\ {}^{E}_{B}R_{21} & {}^{E}_{B}R_{22} & {}^{E}_{B}R_{23} \\ {}^{E}_{B}R_{31} & {}^{E}_{B}R_{31} & {}^{E}_{B}R_{31} \end{bmatrix} \tag{2} $$

Using Euler angles, the rotation matrix can be expressed as

$$ {}^{E}_{B}R = \begin{bmatrix} \theta_c\psi_c & \phi_s\theta_s\psi_c - \phi_c\psi_s & \phi_c\theta_s\psi_c + \phi_s\psi_s \\ \theta_c\psi_s & \phi_s\theta_s\psi_s + \phi_c\psi_c & \phi_c\theta_s\psi_s - \phi_s\psi_c \\ -\theta_s & \phi_s\theta_c & \phi_c\theta_c \end{bmatrix} \tag{3} $$

where ϕ (roll), θ (pitch), and ψ (yaw) are the Euler angles; subscripts c and s stand for cosine and sine of an angle.

The rotation matrix, which is given by (3), is called the DCM. From (2) & (3), three Euler angles can be calculated as follows

$$ \theta = -\sin^{-1}({}^{E}_{B}R_{31}) \tag{4} $$

$$ \phi = \tan^{-1}({}^{E}_{B}R_{32}, {}^{E}_{B}R_{33}) \tag{5} $$

$$ \psi = \tan^{-1}({}^{E}_{B}R_{21}, {}^{E}_{B}R_{11}) \tag{6} $$

Therefore, we estimate elements of the DCM instead of Euler angles directly. This technique can avoid the singularity problem in Euler based method. In practice, we only need estimate two out of three rows of the DCM. A remaining row is calculated from estimated rows based on the DCM orthogonalization property. In this paper, we divide the estimation algorithm into two steps sequentially. In first step, we estimate the third row of the DCM. Then, we calculate ϕ and θ angles using (4) & (5). In second step, we estimate the first row of the DCM. Then, we calculate the second row of the DCM and ψ angle using (6). The main advantage of this technique is the removal the effect of magnetic interference on the roll and pitch angles in comparison with quaternion based method. In addition, this helps reduce computation effort because of decreasing the number of states in Kalman filter.

The kinematic equation of the DCM is expressed as

$$ {}^{E}_{B}\dot{R}(t) = {}^{E}_{B}R(t).[\omega] \tag{7} $$

where ${}^{B}\omega = [\,{}^{B}\omega_x \quad {}^{B}\omega_y \quad {}^{B}\omega_y\,]^T$ is the angular velocity vector, and the skew matrix $[\omega]$ is given by:

$$ [\omega] = \begin{bmatrix} 0 & -{}^{B}\omega_z & {}^{B}\omega_y \\ {}^{B}\omega_z & 0 & -{}^{B}\omega_x \\ {}^{B}\omega_y & {}^{B}\omega_x & 0 \end{bmatrix} $$

Linearize this kinematic equation by first-order approximation, equation (7) becomes

$$ {}^{E}_{B}R_t = (I + [\omega]\Delta T)\,{}^{E}_{B}R_{t-1} \tag{8} $$

This equation is used in process model of the Kalman filter of our algorithm.

B. Modeling of Sensor Signals

Accelerometer: the accelerometer signals on three axes are modeled as the sum of the gravity ${}^{B}a_0$, the external acceleration ${}^{B}a_e$, and a white noise w_a. All three terms are respected to sensor frame. The external acceleration vector ${}^{B}a_e$ is modeled as a low-pass filter process as in [5],

$$ {}^{B}a = {}^{B}a_0 + {}^{B}a_e + w_a \tag{9} $$

$$ {}^{B}\dot{a}_e = -c_a\,{}^{B}a_e + w_{a,e} \tag{10} $$

Refer to rotation matrix in (1), we have a normalized equation between gravity vector ${}^{B}a_0$ and ${}^{E}a_0$ as follows

$$ {}^{B}a_0 = {}^{E}_{B}R^T\,{}^{E}a_0 = {}^{E}_{B}R^T \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^T = {}^{E}_{B}R^T_{3i} \tag{11} $$

Hence, the measurement equation for accelerometer is formed as

$$ {}^{B}a = {}^{E}_{B}R^T_{3i} + {}^{B}a_e + w_a \tag{12} $$

Gyroscope: the gyroscope signals on three axes are described as the sum of the angular velocity ${}^{B}\omega_0$, the offset ${}^{B}\omega_b$, and a white noise w_ω. The offset ${}^{B}\omega_b$ is modeled as first-order Markov process as in [5],

$$ {}^{B}\omega = {}^{B}\omega_0 + {}^{B}\omega_b + w_\omega \tag{13} $$

$$ {}^{B}\dot{\omega}_b = w_{\omega,b} \tag{14} $$

In this paper, we consider the gyroscope signals as an angular velocity in process model. It means that we neglect the offset and the noise term of gyroscope signals. This technique increases the response capable of the estimation algorithm in dynamic conditions.

Magnetometer: the magnetometer signals are modeled as the sum of the earth magnetic field ${}^{B}m_0$, the magnetic disturbance ${}^{B}m_d$, and a white noise w_m. The disturbance is also modeled as low-pass filter as in [5],

$$ {}^{B}m = {}^{B}m_0 + {}^{B}m_d + w_m \tag{15} $$

$$ {}^{B}\dot{m}_d = -c_d\,{}^{B}m_d + w_{m,d} \tag{16} $$

The normalized equation between the earth magnetic field ${}^{B}m_0$ and ${}^{E}m_0$ is given by

$$
\begin{aligned}
{}^{B}m_0 &= {}^{E}_{B}R^{T}\,{}^{E}m_0 \\
&= {}^{E}_{B}R^{T}\begin{bmatrix}\cos(D) & 0 & \sin(D)\end{bmatrix}^{T} \\
&= {}^{E}_{B}R^{T}_{1i}\cos(D) + {}^{E}_{B}R^{T}_{3i}\sin(D)
\end{aligned} \tag{17}
$$

where D is the inclination or dip angle between the intensity vector of the earth magnetic field and the horizontal plane. At our experiment location, the dip angle D is around 7º. Hence, the measurement equation for magnetometer is

$$
{}^{B}m = {}^{E}_{B}R^{T}_{1i}\cos(D) + {}^{E}_{B}R^{T}_{3i}\sin(D) + {}^{B}m_d + w_m \tag{18}
$$

Or it is rewritten to:

$$
{}^{B}m - {}^{E}_{B}R^{T}_{3i}\sin(D) = {}^{E}_{B}R^{T}_{1i}\cos(D) + {}^{B}m_d + w_m \tag{19}
$$

C. Extended Kalman Filter

The kinematic equation of the DCM in (7) can be written in vector form as follows

$$
\dot{x} = f(x) + w(t) \tag{20}
$$

We must linearize this equation to discrete time domain for Kalman implementation because it is nonlinear and in continuous time domain. Let ΔT be the sampling interval. Then, using Jacobian matrix to linearize this equation by first-order approximation, we get the linear and discrete time equation as below

$$
x_{k+1} = x_k + \left.\frac{df}{dx}\right|_{x_k} x_k \Delta T + w_k \tag{21}
$$

Rewrite equation (21) by using the notation Φ_k, the process model is described as

$$
x_{k+1} = \Phi_k x_k + w_k \tag{22}
$$

where

$$
\Phi_k = I + \left.\frac{df}{dx}\right|_{x_k} \Delta T
$$

Equation (19) and (12) are both linear so that the measurement model is given by

$$
z_k = H_k x_k + v_k \tag{23}
$$

Therefore, the algorithm is represented as state equations as below

$$
\begin{cases}
x_{k+1} = \Phi_k\, x_k + w_k \\
z_{k+1} = H_{k+1}\, x_{k+1} + v_{k+1}
\end{cases} \tag{24}
$$

Finally, the procedure of implementation the Kalman filter can be summarized as follows ([1])

1) Compute the a priori state estimate

$$
x_k^- = \Phi_{k-1} x_{k-1}^+ \tag{25}
$$

2) Compute the a priori error covariance matrix

$$
P_k^- = \Phi_{k-1} P_{k-1}^+ \Phi_{k-1}^T + Q_{k-1} \tag{26}
$$

3) Compute the Kalman gain

$$
K_k = P_k^- H^T (H P_k^- H^T + R_{k-1})^{-1} \tag{27}
$$

4) Compute the a posteriori state estimate

$$
x_k^+ = x_k^- + K_k(z_k - H x_k^-) \tag{28}
$$

5) Compute the a posteriori error covariance matrix

$$
P_k^+ = (I - K_k H) P_k^- \tag{29}
$$

III. Adaptive Extended Kalman Filter

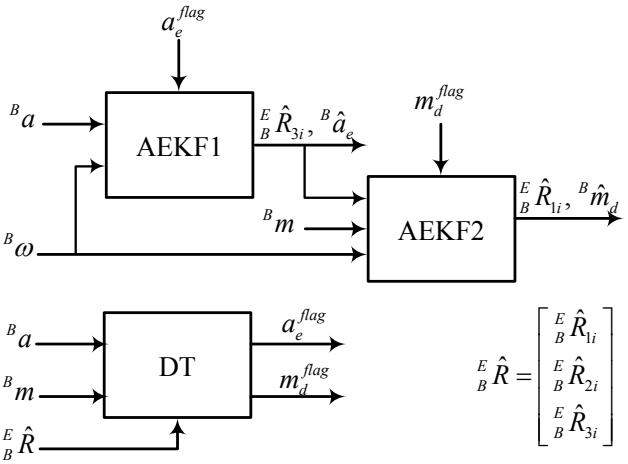

Fig. 1. Model of the estimation algorithm

In our estimation algorithm we use cascade of two adaptive extended Kalman filters and a detector to support filters (Refer to Fig.1). Block AEKF1 in Fig.1 describes the first Kalman filter. It estimates the third row of the DCM and the external acceleration from two sources: accelerometer and gyroscope signals. Block AEKF2 in Fig.1 presents the second Kalman filter. It estimates the first row of the DCM and the magnetic disturbance from three sources that include magnetometer, gyroscope signals, and AEKF1's output. Block DT (Detector) in Fig.1 represents the detector that detects occurrence of the external acceleration and the magnetic disturbance. Then, it generates two flags a_e^{flag}, m_d^{flag} to set or reset the value of variables ${}^{B}a_e$, ${}^{B}m_d$ in filters. This detector ensures the algorithm achieves a better estimation during both static and dynamic conditions. The detail description about each block in our algorithm is described below.

A. AEKF1 Block

The first filter AEKF1 has a state equation as below

$$
\begin{cases}
{}^{1}x_{k+1} = {}^{1}\Phi_k(\Delta T, \omega)\,{}^{1}x_k + {}^{1}w_k \\
{}^{1}z_{k+1} = {}^{1}H_{k+1}\,{}^{1}x_{k+1} + {}^{1}v_{k+1}
\end{cases} \tag{30}
$$

where: ${}^{1}x_k = \begin{bmatrix} {}^{E}_{B}R_{31} & {}^{E}_{B}R_{32} & {}^{E}_{B}R_{33} & {}^{B}a_{ex} & {}^{B}a_{ey} & {}^{B}a_{ez} \end{bmatrix}^T$ is 6x1 vector, ΔT is sample time,

and ${}^{1}z_k = \begin{bmatrix} {}^{B}a_x & {}^{B}a_y & {}^{B}a_z \end{bmatrix}^T$ is 3x1 vector.

The matrix ${}^{1}\Phi_k$ which is the first-order approximation of the (7) and the matrix ${}^{1}H_k$ in measurement equation are expressed as

$$\begin{cases} {}^1\Phi_k = \begin{bmatrix} [\omega\otimes] & 0_3 \\ 0_3 & (1-c_a\Delta T)I_3 \end{bmatrix} \\ {}^1H_k = [I_3 \quad I_3] \end{cases} \tag{31}$$

where

$$\begin{aligned} [\omega\otimes] &= I + [\omega]\Delta T \\ &= \begin{bmatrix} 1 & {}^B\omega_z.\Delta T & -{}^B\omega_y.\Delta T \\ -{}^B\omega_z.\Delta T & 1 & {}^B\omega_x.\Delta T \\ {}^B\omega_y.\Delta T & -{}^B\omega_x.\Delta T & 1 \end{bmatrix} \end{aligned} \tag{32}$$

Covariance matrix of process noise and covariance matrix of measurement noise are described as

$$\begin{aligned} {}^1Q_k &= E\left\{ {}^1w_k\,{}^1w_k^T \right\} \\ &= \begin{bmatrix} \Delta T^2 R_\omega [R_{3i}][R_{3i}]^T & 0_{3x3} \\ 0_{3x3} & q_e I_{3x3} \end{bmatrix} \end{aligned} \tag{33}$$

$${}^1R_k = E\left\{ {}^1v_k\,{}^1v_k^T \right\} = \begin{bmatrix} \sigma_{a_x} & 0 & 0 \\ 0 & \sigma_{a_y} & 0 \\ 0 & 0 & \sigma_{a_z} \end{bmatrix} \tag{34}$$

where

$$\begin{aligned} [R_{3i}] &= \begin{bmatrix} 0 & -{}^E_B R_{33} & {}^E_B R_{32} \\ {}^E_B R_{33} & 0 & -{}^E_B R_{31} \\ -{}^E_B R_{32} & {}^E_B R_{31} & 0 \end{bmatrix} \\ R_\omega &= \begin{bmatrix} \sigma_{\omega_x} & 0 & 0 \\ 0 & \sigma_{\omega_y} & 0 \\ 0 & 0 & \sigma_{\omega_z} \end{bmatrix} \end{aligned} \tag{35}$$

B. AEKF2 Block

The state equation of the second filter AEKF2 is described as below

$$\begin{aligned} {}^2x_{k+1} &= {}^2\Phi_k(\Delta T,\omega)\,{}^2x_k + {}^2w_k \\ {}^2z_{k+1} &= {}^2H_{k+1}\,{}^2x_{k+1} + {}^2v_{k+1} \end{aligned} \tag{36}$$

where

${}^2x_k = \begin{bmatrix} {}^E_B R_{11} & {}^E_B R_{12} & {}^E_B R_{13} & {}^B m_{dx} & {}^B m_{dy} & {}^B m_{dz} \end{bmatrix}^T$ is 6x1 vector, ΔT is sample time, and a 3x1 vector ${}^2z_k = \begin{bmatrix} {}^B m_x & {}^B m_y & {}^B m_z \end{bmatrix}^T - \begin{bmatrix} {}^E_B R_{31} & {}^E_B R_{32} & {}^E_B R_{32} \end{bmatrix}^T \sin(D)$.

The matrix ${}^2\Phi_k$ which is the first-order approximation of the (7) and the matrix ${}^2H_k$ in measurement equation are expressed as

$$\begin{cases} {}^2\Phi_k = \begin{bmatrix} [\omega\otimes] & 0_{3x3} \\ 0_{3x3} & (1-c_d\Delta T)I_{3x3} \end{bmatrix} \\ {}^2H_k = [\cos(D)\,I_3 \quad I_3] \end{cases} \tag{37}$$

Process noise and measurement noise covariance matrix are given by:

$$\begin{aligned} {}^2Q_k &= E\left\{ {}^2w_k\,{}^2w_k^T \right\} \\ &= \begin{bmatrix} \Delta T^2 R_\omega [R_{1i}][R_{1i}]^T & 0_{3x3} \\ 0_{3x3} & q_d I_{3x3} \end{bmatrix} \end{aligned} \tag{38}$$

$${}^2R_k = E\left\{ {}^2v_k\,{}^2v_k^T \right\} = \begin{bmatrix} \sigma_{m_x} & 0 & 0 \\ 0 & \sigma_{m_y} & 0 \\ 0 & 0 & \sigma_{m_z} \end{bmatrix} \tag{39}$$

where

$$\begin{aligned} [R_{1i}] &= \begin{bmatrix} 0 & -{}^E_B R_{13} & {}^E_B R_{12} \\ {}^E_B R_{13} & 0 & -{}^E_B R_{11} \\ -{}^E_B R_{12} & {}^E_B R_{11} & 0 \end{bmatrix} \\ R_\omega &= \begin{bmatrix} \sigma_{\omega_x} & 0 & 0 \\ 0 & \sigma_{\omega_y} & 0 \\ 0 & 0 & \sigma_{\omega_z} \end{bmatrix} \end{aligned} \tag{40}$$

C. DT Block

To detect the occurrence of the external acceleration and the magnetic disturbance, we compare its absolute value after estimation and a threshold. When it exceeds the threshold, the detector turns on the flag signals. On the contrary, the detector turns off the flag signals. The flag signals are 3x1 vectors which respect to each value on three axes of the sensor signals. The detection algorithm in this block is described below

$$\begin{cases} {}^B a_{e,k} \approx {}^B a_k - {}^E_B \hat{R}^T_{3i,k-1} \\ {}^B a^{flag}_{e,k} = \left(\left| {}^B a_{e,k} \right| > \delta_A \right) ?1:0 \end{cases} \tag{41}$$

$$\begin{cases} {}^B m_{d,k} \approx {}^B m_k - {}^E_B \hat{R}^T_{3i,k-1} \sin(D) - {}^E_B \hat{R}^T_{1i,k-1} \cos(D) \\ {}^B m^{flag}_{d,k} = \left(\left| {}^B m_{d,k} \right| > \delta_M \right) ?1:0 \end{cases} \tag{42}$$

Due to the modeling of the sensor signals, the external acceleration ${}^B a_{e,k}$ (at time kT) is almost equal to the difference between the acceleration ${}^B a_k$ and the quantity ${}^E_B \hat{R}^T_{3i,k-1}$ (estimation value at time $(k-1)T$). The flag ${}^B a^{flag}_{e,k}$ is set to 1 if the absolute value of ${}^B a_{e,k}$ is greater than the threshold δ_A. For example, the external acceleration ${}^B a_{e,k}$ is $\begin{bmatrix} 0.3 & 0.05 & -0.2 \end{bmatrix}^T$, and the threshold δ_A is 0.1 so that the flag ${}^B a^{flag}_{e,k}$ can be $\begin{bmatrix} 1 & 0 & 1 \end{bmatrix}^T$. Similarly, the disturbance ${}^B m_{d,k}$ is computed, and then the flag ${}^B m^{flag}_{d,k}$ is generated after comparison with the threshold δ_M.

IV. EXPERIMENTAL RESULT

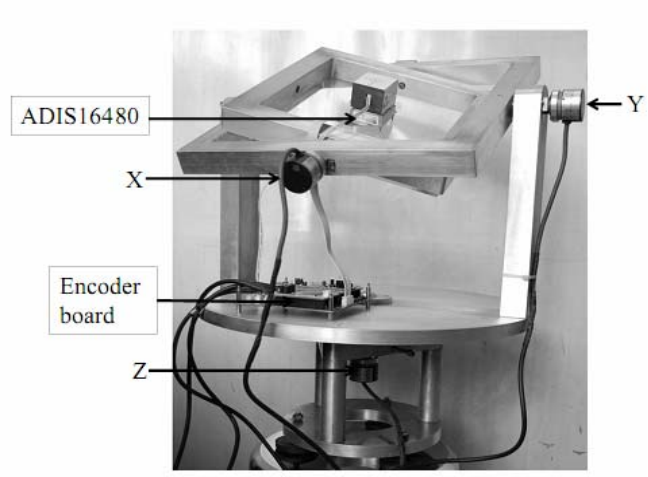

Fig. 2. The rotation table used in the experiment

To verify the performance of the proposed algorithm, we use a commercial IMU and a rotation table. The IMU is ADIS16480 [8] from Analog Devices, its output contains both estimation angles and raw data (accelerometer, gyroscope, and magnetometer). Table 1 highlights its performance based on the datasheet:

Table 1. ADIS16480 Parameters

Parameters	Quantity	Unit
ANGLE OUTPUTS		
Static Accuracy	0.1° (pitch, roll)	Degrees
	0.3° (yaw)	Degrees
Dynamic Accuracy	0.3° (pitch, roll)	Degrees
	0.5° (yaw)	Degrees
Sensitivity	0.0055°	Degrees
GYROSCOPES		
Sensitivity	3.052×10^{-7}	°/sec/LSB
Output noise	0.16	°/sec rms
In-Run Bias Stability	6.25	°/hr
ACCELEROMETERS		
Sensitivity	1.221×10^{-8}	g/LSB
Output noise	1.5	mg rms
In-Run Bias Stability	0.1	mg
MAGNETOMETERS		
Sensitivity	0.1	mgauss/LSB

The IMU is attached to the rotation table which is with three encoders on three axes. Now, we assume that the true angles are calculated from the encoder of the table (x^{enc}). We collect the data from the IMU, the table, and then we use MATLAB to simulate the proposed algorithm. The output angles from MATLAB simulation (the estimation value x^{est}), and angles from the IMU (the reference value x^{ref}) are compared to the true value x^{enc}. Let e^{est}, and e^{ref} be the errors that are described as follows

$$\begin{aligned} e^{est} &= x^{est} - x^{enc} \\ e^{ref} &= x^{ref} - x^{enc} \end{aligned} \tag{43}$$

Using measurement data from the IMU, we calculate the variances for accelerometer and magnetometer signals. From the trial and error experiments, Kalman filter parameters are shown as below:

$$^{1}R_k = \text{diag}([1\text{e-}6\ 1\text{e-}6\ 2\text{e-}5])$$

$$R_\omega = \text{diag}([2\text{e-}6\ 2\text{e-}6\ 6\text{e-}6])$$

$$^{2}R_k = \text{diag}([2\text{e-}7\ 5\text{e-}7\ 1\text{e-}6])$$

$$c_a = 0.1\,,\ \delta_A = 0.1$$

$$c_d = 0.5\,,\ \delta_M = 0.02$$

To carry out a test, we get its response during some operations such as turning over three axes, being stationary…Table 2 shows the testing results.

Table 2. Testing results

N	Description	e^{est}_{RMS} (deg)			e^{ref}_{RMS} (deg)		
		ϕ	θ	ψ	ϕ	θ	ψ
1	Turn x-axis (±90º)	0.94	0.13	0.23	0.63	0.13	0.14
2	Turn y-axis (±70º)	0.07	0.64	0.71	0.07	0.49	0.26
3	Turn z-axis (±180º)	0.15	0.20	0.26	0.11	0.19	0.32
4	Turn 3 axes	0.23	0.18	0.27	0.44	0.23	0.35
5	Standstill (400sec)	0	0.04	0	0	0.06	0.06
6	Standstill with the magnetic disturbance	0	0	0.31	0.52	0.10	3.8

The experimental results show that the proposed algorithm is stable and achieves high precision in all three angles. There is no drift on angles. This shows the feasibility to implement this algorithm to real-time systems.

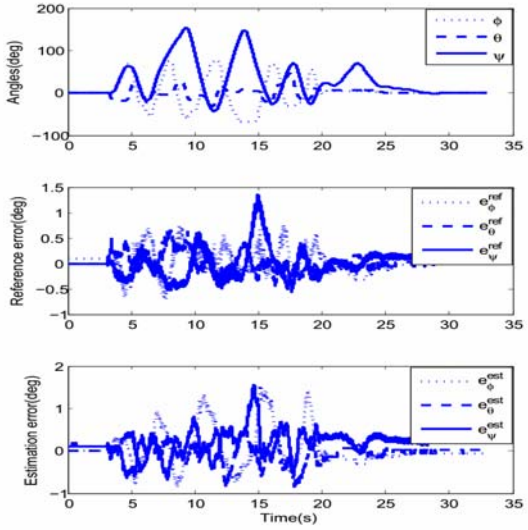

Fig. 3. Manual turning the rotation table on three axes. The RMS error of the reference value is $e^{ref}_{RMS} = [0.23^\circ\ \ 0.18^\circ\ \ 0.27^\circ]$, and the RMS error of the estimation value is $e^{est}_{RMS} = [0.44^\circ\ \ 0.23^\circ\ \ 0.35^\circ]$. The proposed algorithm achieves the good result during dynamic condition.

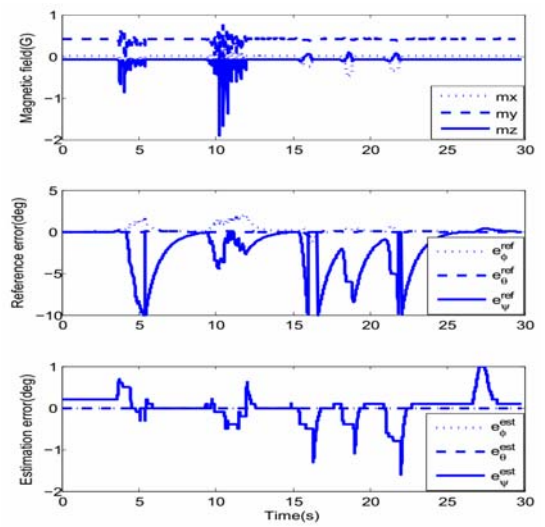

Fig. 4. Let the IMU be stationary, and disturbed the magnetic field to verify the response of the algorithm. The RMS error of the reference value is $e_{RMS}^{ref} = [0.52^\circ \;\; 0.1^\circ \;\; 3.8^\circ]$, and the RMS error of the estimation value is $e_{RMS}^{est} = [0^\circ \;\; 0^\circ \;\; 0.31^\circ]$. Due to using cascade of two Kalman filters, the magnetic disturbance does not affect the roll and pitch angles. The adaptive method keeps the yaw angle in high precision.

V. Conclusion

In this paper, we have presented and implemented the fusion algorithm for orientation estimation with two adaptive extended Kalman filters to enhance the capable of the algorithm in dealing with dynamic conditions. Testing results have shown many advantages of our proposed algorithm in solving original problems of orientation estimation. Further research can be conducted to compare the accuracy and computation load of this algorithm with other approaches such as Euler and quaternion based Kalman filter, or complementary filter.

Acknowledgment

This research is funded by National Science & Technology Foundation of Vietnam (No. KC03.15/11-15) and NAFOSTED under grant number 102.99-2011.25

References

[1] Greg Welch and Gary Bishop (2006). *An Introduction to the Kalman Filter*

[2] Bachmann, E.R. *et al*, "Orientation Tracking for Humans and Robots Using Inertial Sensors", *In 1999 International Symposium on Computational Intelligence in Robotics & Automation,* 1999, pp.187-194.

[3] Marins, J.L., Yun, X., Bachmann, E.R., McGhee, R.B., Zyda, M.J., "An Extended Kalman Filter for Quaternion-Based Orientation Estimation Using MARG Sensors", *In Proceedings of the 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2001, vol.4, pp. 2003-2011.

[4] Yun, X., Bachmann, E.R., "Design, Implementation, and Experimental Results of a Quaternion-Based Kalman Filter for Human Body Motion Tracking", *In IEEE Transactions on Robotics*, 2006, vol. 22, pp. 1216-1227.

[5] Daniel Roetenberg, *Inertial and Magnetic Sensing of Human Motion,* PhD Thesis, 2006, University of Twente.

[6] Nguyen Ho Quoc Phuong, "Study On Orientation Estimation With Three Different Representations". *In Proceedings of the International Symposium on Electrical & Electronics Engineering,* 2007, pp. 80-88, Vietnam.

[7] Madgwick, S.O.H., "Estimation of IMU and MARG orientation using a gradient descent algorithm", *Rehabilitation Robotics (ICORR),* 2011, pp. 1-7.

[8] Analog Devices, ADIS16480 *i*Sensor, http://www.analog.com

[9] Young Soo Suh, "Orientation estimation using a quaternion-based indirect Kalman filter with adaptive estimation of external acceleration", *IEEE Transactions on Instrumentation and Measurement,* 2010, vol. 59 , pp. 3296 – 3305.

[10] Dan Simon, *Optimal State Estimation Kalman, H_∞, and Nonlinear Approaches*, John Wiley & Sons, 2006.