UAV Attitude Estimation Using Unscented Kalman Filter and TRIAD

Hector Garcia de Marina, Fernando J. Pereda, Jose M. Giron-Sierra, *Member, IEEE*, and Felipe Espinosa, *Member, IEEE*

***Abstract*—A main problem in autonomous vehicles in general, and in unmanned aerial vehicles (UAVs) in particular, is the determination of the attitude angles. A novel method to estimate these angles using off-the-shelf components is presented. This paper introduces an attitude heading reference system (AHRS) based on the unscented Kalman filter (UKF) using the three-axis attitude determination (TRIAD) algorithm as the observation model. The performance of the method is assessed through simulations and compared to an AHRS based on the extended Kalman filter (EKF). The paper presents field experiment results using a real fixed-wing UAV. The results show good real-time performance with low computational cost in a microcontroller.**

***Index Terms*—Attitude heading reference system (AHRS), extended Kalman filter (EKF), three-axis attitude determination (TRIAD), unmanned aerial vehicle (UAV), unscented Kalman filter (UKF).**

I. Introduction

THERE IS growing interest in autonomous vehicles. These vehicles are suitable for mobile missions, specially in vigilance, monitoring, and inspection scenarios [1], [2]. Be it ground, marine, or aerial, controlling an autonomous vehicle needs some knowledge of its attitude angles [3]–[5]. These angles can be measured in different ways, for instance, using a conventional inertial navigation system. Modern microelectromechanical systems (MEMS) technologies are offering light and moderate-cost solutions, denoted as inertial measurement units (IMUs), which are appropriate for lightweight unmanned aerial vehicles (UAVs).

Our research group is involved in the development of UAVs relying on experimentation through a spiral life cycle development based on prototypes. An on-board hardware and software have been designed for these UAVs. The hardware includes an IMU with three-axis accelerometers, gyrometers, and magnetometers; a global positioning system (GPS) receiver is also included. The system is light and small enough to fly on a small fixed-wing UAV.

Manuscript received December 24, 2010; revised March 29, 2011; accepted July 11, 2011. Date of publication August 8, 2011; date of current version June 19, 2012.

H. G. de Marina was with the Department of Computer Architecture and Automatics, Complutense University of Madrid, 28040 Madrid, Spain. He is now with the GNC group at Deimos-Space, 28760 Tres Cantos, Spain (e-mail: hgdemarina@gmail.com).

F. J. Pereda and J. M. Giron-Sierra are with the Department of Computer Architecture and Automatics, Complutense University of Madrid, 28040 Madrid, Spain (e-mail: fpereda@gmail.com; gironsi@dacya.ucm.es).

F. Espinosa is with the Electronics Department, University of Alcala de Henares, 28806 Alcala de Henares, Spain (e-mail: felipe@depeca.uah.es).

Digital Object Identifier 10.1109/TIE.2011.2163913

The first flights, using manual control, have been used to gather signals from the sensors. We are looking at closing autonomous control loops, which need an accurate estimation of the three attitude angles. The signals from the sensors have lots of vibrations due to the mechanical nature of the system and noise with bias due to the sensors themselves and environmental effects. A common approach to multisensor data fusion is to employ a Kalman filter.

Conventional kinematic models of flying vehicles are highly nonlinear; the filter should be able to cope with these nonlinearities. A typical Kalman filter for nonlinear systems is the extended Kalman filter (EKF). Using our experimental signals, the performance of the EKF was not completely satisfactory. Sometimes, estimation errors are too high and sometimes the filter may diverge. This is inappropriate for control loops [6].

Searching for a better alternative, we selected the unscented Kalman filter (UKF) as a potentially better solution. While the EKF is based on the linearization of the model through Jacobians or Hessians, the UKF uses the nonlinear model directly; therefore, the predictions should be more accurate.

The nine terms of the direct cosine matrix (DCM) can be measured by using the three-axis attitude determination (TRIAD) algorithm and data from on-board sensors: accelerometers and magnetometers. Using a quaternion formulation, which is a conventional way to deal with the attitude of aerial vehicles [7], the DCM terms can be easily handled. The quaternion approach is widely used in attitude heading reference systems (AHRSs) because it avoids the gimbal lock problem. With respect to numerical stability, quaternions are easier to propagate than the angles themselves.

For systematic and nonrisky work, a simulation environment has been developed. The core of the simulation is the X-Plane 9 simulator which is certified by the Federal Aviation Administration for subsonic terrestrial flight. For realism, perturbations in the form of high-frequency noise and sensor latencies are included to the data from the simulator.

To evaluate the proposed solution, a number of simulations have been run. These have been used to both validate the system and to compare it to the same algorithm using the EKF. During these simulations a maximum error of 1.0° is imposed on the pitch and roll attitude angles, and a maximum error of 4.0° on the yaw attitude angle. Under those requirements, which are standard in the industry [8], the error tolerances in the different signals are obtained through the analysis of a large number of simulation cases. The solution based on the UKF shows better performance.

Fig. 1. UAV developed by the authors.

Finally, the UKF-based solution is tested on a real experiment using a fixed-wing UAV of 2 m of wingspan. This UAV is shown in Fig. 1. An in-house autopilot is used to test the estimation algorithm presented in this paper. The results of the solution are compared to different independent systems with very good results.

Summarizing, the first section of the paper presents some of the state of the art of AHRSs. The following section is devoted to the formulation of the problem, including the kinematic model and the TRIAD algorithm. Then comes a section describing the whole solution using the UKF and TRIAD. The fifth section deals with data obtained by simulation, and the comparison between the UKF and the EKF is presented. Then, field experiments are presented to fully evaluate the proposed AHRS. In the final section of the paper, some conclusions are drawn, including some guidelines for future work.

II. Background and Related Work

The unscented transformation (UT) and the UKF were introduced by Julier and Uhlmann [9], the algorithms were further explained, with examples, by Wan and Merwe [10]. An extensive description of the UKF was given in [11].

Background on the EKF algorithm is given in [12]. In [13], it was shown that the EKF can lead to unbounded estimation errors for nonlinear systems similar to those used in this paper.

Some modern techniques, such as particle filters [14], are expensive in computational terms. Therefore, they are unsuitable for microcontrollers unless the number of particles is reduced. However, this can lead to worse performance than the EKF [15].

In a recent article [16], a comparison of both the EKF and UKF is done for the particular case of flight path reconstruction for a fixed-wing UAV. This article includes an interesting discussion of previous references that illustrate the improved performance of the UKF over EKF, particularly when used to estimate the attitude of an UAV.

For real-time applications, it is important to reduce the computational cost. Part of the research deals with different ways of alleviating this cost. For instance, in [17], an attempt to simplify the observation model is shown; however, it compromises computational cost because the method involves the inversion of several large matrices every filter iteration. Another example is [18], which proposes to reduce the number of vector state variables. Our proposed solution has been implemented in a microcontroller for real-time estimation.

As highlighted by [19], it is important to base the Kalman filter on an accurate model. In our context, the models presented in the following section are well established in the literature and correspond with experimental results.

An extensive review of navigation systems is [20]. It covers different algorithms including Kalman filters and the TRIAD algorithm. The TRIAD algorithm was introduced by Shuster and Oh in [21] to measure the DCM of a spacecraft.

The use of IMUs in the industry has been increasing in the recent years, in part due to the reduced cost of those based in MEMS technology. In [22], a MEMS-based IMU is used in the tracking system of a fastening tool.

A well-known problem with gyrometers is bias. Therefore, different sensors have to be used to correct these biases. For instance, in [23] they are corrected using three-axis accelerometers. In [24], the use of eight accelerometers in a new configuration is proposed for measuring angular velocities in small UAVs.

Other approaches rely on magnetometers to estimate the yaw angle in helicopters [25]; or in GPS to estimate both the position and the attitude in fixed-wing UAVs [26]. Alternatively, other papers propose not to use gyrometers at all in conventional aircrafts, but several GPS receivers instead [27].

In this paper, we propose an attitude estimator using the UKF and the TRIAD algorithm. Gyrometers, magnetometers, and accelerometers are involved in the proposed estimator. The validation of the algorithm is done by both simulations and field experiments. Experiments use an on-board hardware with MEMS sensors.

III. Problem Formulation

Although each UAV has its own coefficients and therefore its own dynamical model, it is possible to determine Euler angles from a kinematic model, which is independent of the coefficients of the UAV. In this section, we derive the mathematical formulation of the AHRS problem in a UAV equipped with a three-axis gyrometer, three-axis accelerometer, and three-axis magnetometer.

A. AHRS Kinematic Model

The Euler angles describe the aircraft body-axis orientation in North, East, and down coordinates, that means in longitudinal, lateral, and normal coordinates, with respect to the local tangent plane to the earth and true North. Here, θ is the pitch angle, ϕ the roll angle, and ψ is the yaw angle; as illustrated in Fig. 2. The angular velocity vector expressed in body frame is P for the roll rate, Q is the pitch rate, and R is the yaw rate, and it is related to the earth frame by the transformation given by the kinematics

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & \tan\theta\sin\phi & \tan\theta\cos\phi \\ 0 & \cos\phi & -\sin\phi \\ 0 & \frac{\sin\phi}{\cos\theta} & \frac{\cos\phi}{\cos\theta} \end{bmatrix} \begin{bmatrix} P \\ Q \\ R \end{bmatrix}. \quad (1)$$

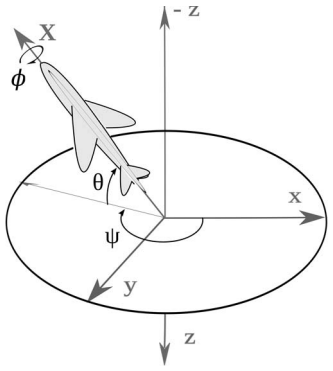

Fig. 2. Axes and coordinate definitions.

Integrating equation (1) gives numerical instability and could be gimbal locked. For this reason, a quaternion formulation to represent the attitude is preferred

$$q = q_0 + q_1 i + q_2 j + q_3 k \qquad \sum_{i=0}^{3} q_i^2 = 1 \tag{2}$$

where the quaternion norm is 1 and its components from Euler angles are

$$q_0 = \cos\phi' \cos\theta' \cos\psi' + \sin\phi' \sin\theta' \sin\psi' \tag{3}$$

$$q_1 = \sin\phi' \cos\theta' \cos\psi' - \cos\phi' \sin\theta' \sin\psi' \tag{4}$$

$$q_2 = \cos\phi' \sin\theta' \cos\psi' + \sin\phi' \cos\theta' \sin\psi' \tag{5}$$

$$q_3 = \cos\phi' \cos\theta' \sin\psi' - \sin\phi' \sin\theta' \cos\psi' \tag{6}$$

where $\phi' = \phi/2$, $\theta' = \theta/2$, and $\psi' = \psi/2$.

The kinematics equation (1) can be rewritten in linear form using quaternion components

$$\begin{bmatrix} \dot{q_0} \\ \dot{q_1} \\ \dot{q_2} \\ \dot{q_3} \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 0 & -P & -Q & -R \\ P & 0 & R & -Q \\ Q & -R & 0 & P \\ R & Q & -P & 0 \end{bmatrix} \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix}. \tag{7}$$

Additionally, it is useful to formulate the DCM using quaternion components and the Euler angles from the DCM terms

$$\begin{aligned} \text{DCM} \equiv A &= \{c_{ij}\} \\ &= \begin{bmatrix} A_1 & 2(q_1q_2+q_0q_3) & 2(q_1q_3-q_0q_2) \\ 2(q_1q_2-q_0q_3) & A_2 & 2(q_2q_3+q_0q_1) \\ 2(q_1q_3+q_0q_2) & 2(q_2q_3-q_0q_1) & A_3 \end{bmatrix} \end{aligned} \tag{8}$$

where $A_1 = q_0^2 + q_1^2 - q_2^2 - q_3^2$, $A_2 = q_0^2 - q_1^2 + q_2^2 - q_3^2$, and $A_3 = q_0^2 - q_1^2 - q_2^2 + q_3^2$. Then,

$$\theta = -\arcsin\left(2(q_1q_3 - q_0q_2)\right) \tag{9}$$

$$\phi = \text{atan2}\left(2(q_2q_3 + q_0q_1), q_0^2 - q_1^2 - q_2^2 + q_3^2\right) \tag{10}$$

$$\psi = \text{atan2}\left(2(q_1q_2 + q_0q_3), q_0^2 + q_1^2 - q_2^2 - q_3^2\right) \tag{11}$$

where atan2 is the four-quadrant version of the inverse tangent function, and arcsin is the arcsine function.

B. Gyros Integration Problem

The three-axis gyrometer measures the angular velocities, and for obtaining the Euler angles, the gyros can be integrated using equation (7). However, even if we ignore the sensor noise, the gyros usually have bias, making their integration error grow in every step.

Fortunately, for a MEMS gyrometer in normal conditions (not extremal temperature or pressure variation), this bias can be assumed to be constant [28], or very slow varying throughout the UAV mission. Therefore, the bias for the gyrometer can be modeled as

$$\mathbf{b} = [b_x \quad b_y \quad b_z]^T \quad \text{with} \quad \dot{\mathbf{b}} = 0. \tag{12}$$

Denoting the angular velocity vector $\boldsymbol{\omega} = [P \ Q \ R]^T$, if the angular velocities from the gyros are $\boldsymbol{\omega}_s$, they can be corrected using the bias as

$$\boldsymbol{\omega} = \boldsymbol{\omega}_s - \mathbf{b}. \tag{13}$$

Another issue arises when equation (7) is numerically integrated. The quaternion shall conserve its norm equals to 1, otherwise the Euler angles are wrongly computed from the quaternion. Although it can be normalized after every integration step, there is a better way to do it. The quaternion norm can be kept by solving equation (7) using the next integrating factor

$$\exp\left(\int_{t_0}^{t} \Omega \, dt\right) \tag{14}$$

where

$$\Omega = \frac{1}{2} \begin{bmatrix} 0 & -P & -Q & -R \\ P & 0 & R & -Q \\ Q & -R & 0 & P \\ R & Q & -P & 0 \end{bmatrix}. \tag{15}$$

Assuming that the angular velocities remain constant during the interval $\mathrm{d}t$, we can discretize equation (7) as follows [29]:

$$q(k+1) = \left(I \cos\frac{\|\Delta\boldsymbol{\omega}\|}{2} + \sin\frac{\|\Delta\boldsymbol{\omega}\|}{2} \|\Delta\boldsymbol{\omega}\| \Omega\right) q(k) \tag{16}$$

where $\|\Delta\boldsymbol{\omega}\| = (1/2)\sqrt{(P\Delta t)^2 + (Q\Delta t)^2 + (R\Delta t)^2}$ and I is the 4×4 identity matrix.

C. TRIAD Algorithm

The TRIAD algorithm was introduced by Shuster and Oh [21] to determine the attitude in a spacecraft from a set of vector measurements. TRIAD is a deterministic method to compute the DCM. Given the reference unit vectors $\mathbf{V_1}$ and $\mathbf{V_2}$ and the corresponding observation unit vectors $\mathbf{W_1}$ and $\mathbf{W_2}$, the DCM satisfies

$$A\mathbf{V_1} = \mathbf{W_1} \qquad A\mathbf{V_2} = \mathbf{W_2}. \tag{17}$$

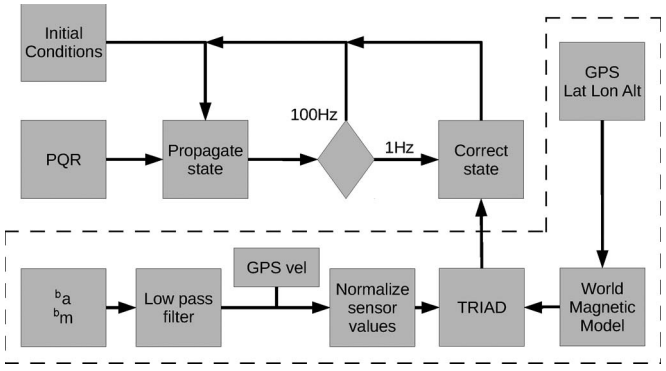

Fig. 3. Algorithm block diagram.

The TRIAD algorithm determines the DCM using the following expression:

$$A = M_o M_r^T \tag{18}$$

$$M_o = (\mathbf{o_1} \,|\, \mathbf{o_2} \,|\, \mathbf{o_3}) \tag{19}$$

$$M_r = (\mathbf{r_1} \,|\, \mathbf{r_2} \,|\, \mathbf{r_3}) \tag{20}$$

where the observation column vectors $\mathbf{o_i}$ and the reference column vectors $\mathbf{r_i}$ are given by

$$\mathbf{o_1} = \mathbf{W_1} \tag{21}$$

$$\mathbf{o_2} = (\mathbf{W_1} \times \mathbf{W_2})/|\mathbf{W_1} \times \mathbf{W_2}| \tag{22}$$

$$\mathbf{o_3} = (\mathbf{W_1} \times (\mathbf{W_1} \times \mathbf{W_2}))\,/|\mathbf{W_1} \times \mathbf{W_2}| \tag{23}$$

$$\mathbf{r_1} = \mathbf{V_1} \tag{24}$$

$$\mathbf{r_2} = (\mathbf{V_1} \times \mathbf{V_2})/|\mathbf{V_1} \times \mathbf{V_2}| \tag{25}$$

$$\mathbf{r_3} = (\mathbf{V_1} \times (\mathbf{V_1} \times \mathbf{V_2}))\,/|\mathbf{V_1} \times \mathbf{V_2}|. \tag{26}$$

Notice that the pair $(\mathbf{W_1}, \mathbf{V_1})$ has more influence on A than $(\mathbf{W_2}, \mathbf{V_2})$, this is because part of the information contained in the second pair is discarded. Therefore, it is convenient to assign $(\mathbf{W_1}, \mathbf{V_1})$ to the pair of greater accuracy, which depends on the flight circumstances. Section IV-B describes criteria to assign these pairs.

IV. Unscented Kalman Filter Design

The UKF [9] is an alternative to the EKF, providing superior performance at similar order of computational cost. Also, during the UKF implementation process, there are not Jacobians, Hessians, or other derivatives involved. In this paper, the UKF is used both for state estimation (quaternion components) and for parameter estimation (gyrometer bias).

The estimation algorithm is implemented as a two-step *propagator/corrector* filter. It is desirable to run each step as many times as possible; however, the frequencies at which they will be run will be limited by different factors. For the propagation step, the limiting factor is the computation time; we chose a frequency of 100 Hz. The correction step is limited, in our case, by the GPS sample rate, that is, 1 Hz.

The algorithm is described in the diagram shown in Fig. 3. The diagram depicts the two main loops, one at 100 Hz and the other at 1 Hz; these loops correspond to the propagation and correction steps, respectively. It can be noted how the TRIAD information feeds the correction step.

The area surrounded by dashed lines contains the elements involved in the TRIAD calculations. The leftmost block represents an access to the sensors to measure ${}^b a$ and ${}^b m$ which correspond to the accelerations and magnetic field measurements in the body reference frame, respectively. The signals are then low-pass filtered. The GPS velocity is used to subtract the centrifugal accelerations as explained in Subsection B, which also covers the description of the TRIAD block. On the right side of the diagram, the block denoted as *World Magnetic Model* uses a harmonic spherical model to obtain the magnetic field vector at the position of the UAV. This vector is used as one of the references in the TRIAD algorithm.

This section describes the design of the filter under the considerations given above. Subsection A gives the equations involved in the propagation loop while Subsection B deals with those of the correction loop.

A. Propagation Equations and Process Model

This subsection describes the propagation loop. The state vector is

$$x(k) = [\, q_0(k) \quad q_1(k) \quad q_2(k) \quad q_3(k) \quad b_x(k) \quad b_y(k) \quad b_z(k) \,]^T \tag{27}$$

where q_i are the quaternion components and b_j are the gyrometers biases. These are assumed to be Gaussian random variables. Their process model is given by (16) and (12), respectively.

The basis of the UKF is the UT, which is a method for calculating the statistics of a random variable which undergoes a nonlinear transformation [9]. In our case, we assume that the noise of the UAV sensors is additive (zero mean). Under this assumption, the equations to estimate the Euler angles and the gyrometers biases are the ones described in this subsection.

The computation algorithm begins with the initial conditions

$$\hat{x}_0 = E[x_0] \tag{28}$$

$$P_0 = E\left[(x_0 - \hat{x}_0)(x_0 - \hat{x}_0)^T\right]. \tag{29}$$

At the start of every iteration, we calculate the *sigma points* [9] χ_{k-1} of the quaternion components and gyrometers biases

$$\chi_{k-1} = [\, \hat{x}_{k-1} \quad \hat{x}_{k-1} + \gamma\sqrt{P_{k-1}} \quad \hat{x}_{k-1} - \gamma\sqrt{P_{k-1}} \,] \tag{30}$$

where $\gamma = \sqrt{L + \lambda}$, λ is a composite scaling parameter and L is the state vector dimension. $\sqrt{P_k}$ can be computed with the lower triangular Cholesky factorization.

The next step is to evaluate the model on the sigma points

$$\chi^*_{k|k-1} = F[\chi_{k-1}, u_{k-1}] \tag{31}$$

where F is the process model as given by equations (16) and (12). u_{k-1} is the angular velocity vector $\boldsymbol{\omega}$. The *a priori* state estimation is approximated using a weighted sample mean

$$\hat{x}_k^- = \sum_{i=0}^{2L} W_i^{(m)} \chi^*_{i,k|k-1} \tag{32}$$

and the covariance matrix is

$$P_k^- = \sum_{i=0}^{2L} W_i^{(c)} \left[\chi^*_{i,k|k-1} - \hat{x}_k^-\right]\left[\chi^*_{i,k|k-1} - \hat{x}_k^-\right]^T + Q_{\text{noise}} \quad (33)$$

where

$$W_0^{(m)} = \lambda/(L+\lambda) \quad (34)$$

$$W_0^{(c)} = \lambda/(L+\lambda) + (1-\alpha^2+\beta) \quad (35)$$

$$W_i^{(m)} = W_i^{(c)} = 1/\left(2(L+\lambda)\right). \quad (36)$$

The constant α determines the spread of the sigma points around $\hat{x}$ and is usually set to a small positive value (for instance, $0 \leq \alpha \leq 1$). The constant β can be used to incorporate prior knowledge of the distribution of the state vector: for Gaussian distributions, $\beta = 2$ is optimal [10]. $Q_{\text{noise}} \in \mathbb{R}^{7x7}$ is the process noise covariance

$$Q_{\text{noise}} = \left[\begin{array}{c|c} Q_q & 0 \\ \hline 0 & Q_b \end{array}\right] \quad (37)$$

where $Q_q \in \mathbb{R}^{4x4}$ is the noise covariance associated to the quaternion components and $Q_b \in \mathbb{R}^{3x3}$ is the noise covariance associated to the gyrometer biases.

The process that relates the quaternion components and angular velocities has a continuous-time analytical solution, as was shown in Section III-B. However, the discrete-time equation (16) assumes that the angular velocities remain constant during the discretization period. Hence, Q_q should be close, but different to zero. For the simulation and experimental results shown in this paper

$$Q_q = 1 \times 10^{-6} \cdot I_{4x4} \quad (38)$$

where I_{4x4} is the 4×4 identity matrix.

The process noise covariance associated to the gyrometer biases Q_b is the 3×3 zero matrix. The rationale for this is the same that explains (12) in Section III-B.

B. Correction Equations and Observation Model

Even though the TRIAD algorithm gives the nine terms of the DCM [see (8)], only four of them are needed to calculate the Euler angles. Hence, the observation function has been designed to measure these four terms.

To determine the pitch and roll angles, the terms are the X and Y components of the Z earth vector expressed in the body frame

$${}^bZ_{Ex} \equiv c_{13} = 2(q_1q_3 - q_0q_2) \quad (39)$$

$${}^bZ_{Ey} \equiv c_{23} = 2(q_2q_3 + q_0q_1). \quad (40)$$

For the yaw angle, the terms are the X and Y components of the X body vector expressed in the earth reference frame

$${}^EX_{bx} \equiv c_{11} = \left(q_0^2 + q_1^2 - q_2^2 - q_3^2\right) \quad (41)$$

$${}^EX_{by} \equiv c_{12} = 2(q_1q_2 + q_0q_3). \quad (42)$$

Therefore, the observation model is given by the following equation:

$$H(x_k) = [c_{13} \quad c_{23} \quad c_{11} \quad c_{12}]^T. \quad (43)$$

According to the TRIAD algorithm, two vector pairs are needed to compute the terms of the DCM. Each pair consists of a measure or observation and a reference vector [see (17)]. In our case, these pairs are the magnetic field, and the acceleration of the UAV.

The magnetic observation vector is the field measured at the body frame. The magnetic reference vector is the earth's magnetic field obtained using a harmonic spherical model; the geographical coordinates for this model are provided by an on-board GPS receiver.

For the acceleration pair, the observation is given by the on-board IMU. It should be noted that the measurement given by the accelerometers includes: linear acceleration, Coriolis acceleration, centripetal acceleration, and gravity. Coriolis acceleration is assumed to be negligible; in our experiments, it is of the order of 10^{-4} g. The reference is the earth's gravity $\hat{\mathbf{g}} = [0\ 0\ 1]^T$, normalized in North East Down coordinates. The on-board accelerometers are affected by mechanical vibrations and environmental perturbations, therefore, it is convenient to use a low-pass filter to reduce these effects. Also, the centrifugal contribution of the acceleration has to be subtracted

$${}^Ea_N = \dot{U} + QW - RV \quad (44)$$

$${}^Ea_E = \dot{V} + RU - PW \quad (45)$$

$${}^Ea_D = \dot{W} + PV - RU \quad (46)$$

where the speed ${}^b\mathbf{V} = [U\ V\ W]^T$ is expressed in the body frame of reference. In a fixed-wing UAV, during a non-acrobatic flight, both V and W are negligible. Therefore, the GPS speed measurement gives U, which is the dominant component of ${}^b\mathbf{V}$.

As stated in Section III-C, it is convenient to assign $(\mathbf{W_1}, \mathbf{V_1})$ to the pair of greater accuracy. During a flight, there are times when accelerometers offer more accuracy, whereas during others, magnetometers are more reliable. In our case, we used the following criteria. In order of priority:

- If $0.9\hat{\mathbf{g}} \leq |{}^b\mathbf{a}| \leq 1.1\hat{\mathbf{g}}$, then, it can be assumed that this is a stationary flight; hence, the acceleration is the pair of greater accuracy. In this case, $\mathbf{V_1} = \hat{\mathbf{g}}$ and $\mathbf{W_1} = {}^b\mathbf{a}$.
- Else, if $0.7\hat{\mathbf{g}} < |{}^b\mathbf{a}| < 0.9\hat{\mathbf{g}}$ or $1.1\hat{\mathbf{g}} < |{}^b\mathbf{a}| < 1.3\hat{\mathbf{g}}$, then, it can be assumed that the plane is doing a coordinated turn. In this case, magnetometers offer greater accuracy. Therefore, $\mathbf{V_1} = {}^E\mathbf{m}$ and $\mathbf{W_1} = {}^b\mathbf{m}$.
- Else, if $|{}^b\mathbf{m}| > 1.2|{}^E\mathbf{m}|$ or $|{}^b\mathbf{m}| < 0.8|{}^E\mathbf{m}|$, then, the magnetic measurements are not reliable. The correction step is skipped.
- Else, if $|{}^b\mathbf{a}| > 1.3\hat{\mathbf{g}}$ or $|{}^b\mathbf{a}| < 0.7\hat{\mathbf{g}}$, then, it is assumed that the current state is acrobatic and neither the magnetic measurements nor the acceleration measurements are reliable enough. The correction step is skipped.

Where ${}^E\mathbf{m}$ is the earth's magnetic field vector and $\hat{\mathbf{g}}$ the earth's gravity acceleration. Notice how in the last two cases, the correction step is skipped. The following only applies in the first two cases.

The UKF begins the correction step by redrawing the sigma points. This is done to incorporate the effect of the additive noise [10]

$$\chi_{k|k-1} = \left[\hat{x}_k^- \quad \hat{x}_k^- + \gamma\sqrt{P_k^-} \quad \hat{x}_k^- - \gamma\sqrt{P_k^-}\right]. \tag{47}$$

Then, the UT of the observations is computed

$$\mathcal{Y}_{k|k-1} = H[\chi_{k|k-1}] \tag{48}$$

$$\hat{y}_k^- = \sum_{i=0}^{2L} W_i^{(m)} \mathcal{Y}_{i,k|k-1}. \tag{49}$$

The measure and cross-covariance matrices are

$$P_{\bar{y}_k\bar{y}_k} = \sum_{i=0}^{2L} W_i^{(c)} \left[\mathcal{Y}_{i,k|k-1} - \hat{y}_k^-\right]\left[\mathcal{Y}_{i,k|k-1} - \hat{y}_k^-\right]^T + R_{\text{noise}} \tag{50}$$

$$P_{x_k y_k} = \sum_{i=0}^{2L} W_i^{(c)} \left[\chi_{i,k|k-1} - \hat{x}_k^-\right]\left[\mathcal{Y}_{i,k|k-1} - \hat{y}_k^-\right]^T \tag{51}$$

where R_{noise} is the measurement noise covariance. The value of R_{noise} can be derived from the nominal values of the errors of the sensors involved. This derivation is described thoroughly in [21].

Now, the Kalman gain is computed and the correction is made

$$\mathcal{K}_k = P_{x_k y_k} P_{\bar{y}_k\bar{y}_k}^{-1} \tag{52}$$

$$\hat{x}_k = \hat{x}_k^- + \mathcal{K}_k\left(y_k - \hat{y}_k^-\right) \tag{53}$$

$$P_k = P_k^- - \mathcal{K}_k P_{\bar{y}_k\bar{y}_k} \mathcal{K}_k^T. \tag{54}$$

V. Simulations Results

Since real experiments might imply crashes, some previous simulations are in order. For this, we developed a simulation framework, which consists in three different parts. The core of the simulation is the X-Plane 9 software. The other two parts are: the plug-in code for X-Plane 9 and the model of the sensors. The idea is to integrate a six-degree-of-freedom aerodynamic model, provided by X-Plane with a realistic model of the sensors we are using. The plug-in code is just the glue between them.

X-Plane 9 includes different aircraft models. Its default radio control model is very similar to our UAV, so no modifications to it are needed. That is the model used in the simulations of this section.

The purpose of the simulations is to study the effects of sensor noise, bias, and latencies. Therefore, the model of the sensors focuses on the following aspects:

- GPS signal is delayed 1 s;
- signals from the gyrometers are biased and corrupted with white noise;
- accelerometers are biased and corrupted with colored, high frequency noise;
- magnetometers are biased and corrupted with white noise.

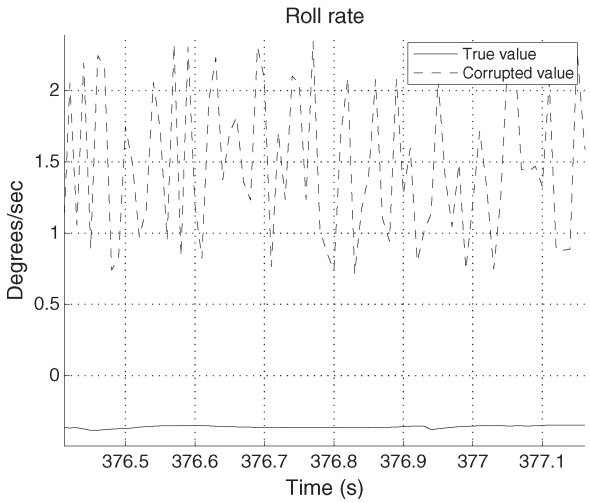

Fig. 4. True and corrupted roll rate used for simulations.

TABLE I
UKF Tolerances: Maximum Bias and Random Error Standard Deviation

Measurement	Bias error	Random error
Roll rate, P	± 9.98 °/sec	± 8.51 °/sec
Pitch rate, Q	± 9.98 °/sec	± 8.51 °/sec
Yaw rate, R	± 9.98 °/sec	± 5.01 °/sec
Accelerometers	± 0.3 m/s^2	± 0.7 m/s^2
Magnetometers	± 10.53mG	± 21.73mG
GPS Velocity	± 2.57m/s	± 2.45m/s

Fig. 4 shows an example of how bias and noise are added to a variable. Notice how both the magnitude of the noise and the bias are kept constant throughout the simulation. In particular, the example shows roll rate measurements as simulated.

A first target of the simulations is to study the tolerances of both estimators to both bias and noise magnitude. According to standard procedures, a maximum error of 1.0° in the estimation of pitch and roll angle, and 4.0° for yaw angle are imposed. It is assumed that with this error, it is possible to do closed-loop control. This is covered in the first part of this section. The second part of this section assesses the performance of the UKF using real values for the biases and noise magnitudes of our sensors.

A. Error Tolerances and Comparison of Estimators

A Monte Carlo analysis of the tolerances was made, supported by a batch of simulation experiments. Each of the experiments specifies different values of biases and noise magnitudes, which were drawn from a Gaussian distribution.

Two sets of results were obtained, one using the UKF and the other using the EKF. These sets are shown in Tables I and II, respectively.

It can be noted how both algorithms are more sensitive to errors in R than they are to errors in P and Q. This is because the information in the yaw angle is only provided by one of the sensors, the magnetometer. While the information in pitch

TABLE II
EKF TOLERANCES: MAXIMUM BIAS AND RANDOM ERROR STANDARD DEVIATION

Measurement	Bias error	Random error
Roll rate, P	± 1.50 °/sec	± 1.50 °/sec
Pitch rate, Q	± 1.50 °/sec	± 1.50 °/sec
Yaw rate, R	± 0.50 °/sec	± 1.50 °/sec
Accelerometers	± 0.14 m/s^2	± 0.32 m/s^2
Magnetometers	± 5.58mG	± 11.43mG
GPS Velocity	± 2.17m/s	± 1.45m/s

TABLE III
TYPICAL MEMS BIAS AND RANDOM ERROR STANDARD DEVIATION

Measurement	Bias error	Random error
Roll rate, P	± 3.00 °/sec	± 1.00 °/sec
Pitch rate, Q	± 3.00 °/sec	± 1.00 °/sec
Yaw rate, R	± 3.00 °/sec	± 1.00 °/sec
Accelerometers	± 0.05 m/s^2	± 0.009 m/s^2
Magnetometers	± 4.00mG	± 1.25mG
GPS Velocity	± 0.5m/s	± 1.5m/s

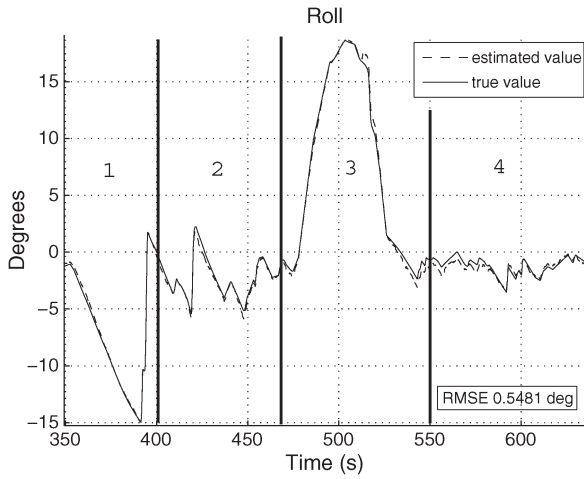

Fig. 5. Roll angle simulated with MEMS parameters.

and roll angles are provided by the accelerometer and the magnetometer.

In general, EKF is more sensitive to biases and noises than the UKF. In particular, it is very sensitive to the bias in R. Hence, the solution based on the UKF is preferred.

B. Simulation Using Real Error Values

For this simulation, the real biases and noise magnitudes were extracted from the datasheet of our IMU. Table III shows them. It is interesting to compare these values to those from Tables I and II. It can be seen how some errors are outside of the EKF tolerances. Hence, EKF cannot be used in our case.

Figs. 5–7 show the behavior of the UKF algorithm in a typical flight in a windy scenario. Solid lines depict simulated angles while the dashed ones depict estimations from the AHRS. It should be noted that in Fig. 7, the abrupt changes near seconds 520 and 590 are due to the representation of angles

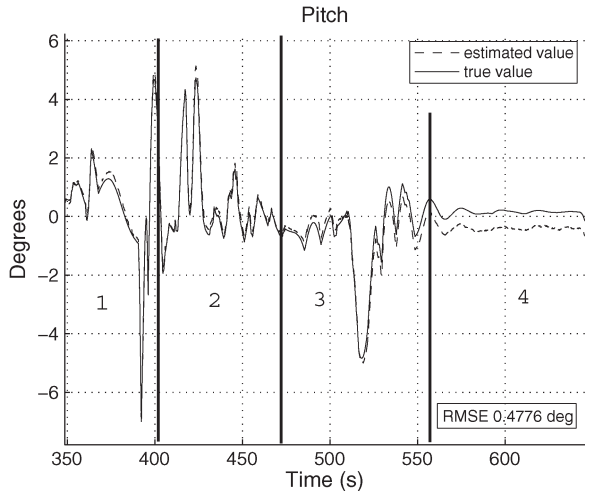

Fig. 6. Pitch angle simulated with MEMS parameters.

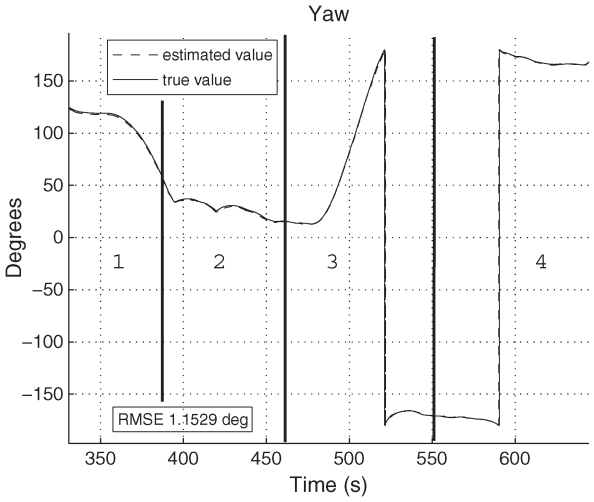

Fig. 7. Yaw angle simulated with MEMS parameters.

in the range $\pm 180°$. The agreement between the simulated and estimated values is satisfactory.

Area marked with "1" shows a coordinated turn to the left, note how the yaw angle decreases. This turn leads to the next area, marked with "2", which is a steady flight. The area marked with "3" is a new coordinated turn to the right, note how the yaw angle increases. The final area, marked with "4", is a new steady flight; the three angles remain mostly constant. It can be noted that the pitch angle estimation in area "4" has a slight bias (of less than 0.3°), this is due to the magnetometers biases added during the data corruption. These biases are not compensated through the algorithm because of their negligible effect compared to the gyrometers biases. The pitch and roll angles experience vibrations due to different gusts of wind.

During the simulations, although it is not very clear in the graphs, it is observed that the proposed solution does not introduce big delays. The time constant of the filter is on the order of 1/100 s.

This section has shown how the simulation results give confidence on the AHRS using the UKF. The system can be used in field experiments. The following section covers field experiments and the validation of the AHRS.

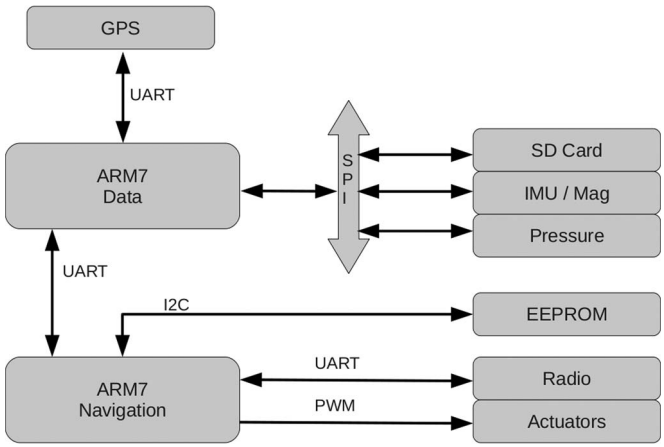

Fig. 8. Functional diagram of the autopilot.

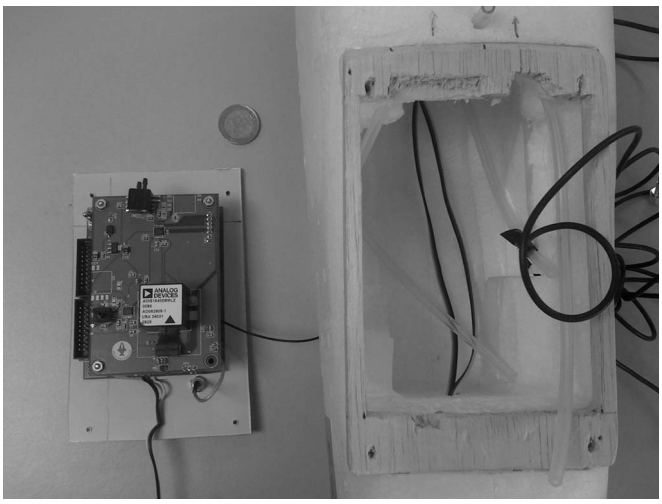

Fig. 9. Photograph of the on-board hardware next to the UAV.

VI. Field Experiment Results

The target of the research is to use the AHRS to feed a closed-loop controller in field experiments. Aside from the simulation results presented in an earlier section, an experimental validation of the system has been done.

Experiments were carried out using a small fixed-wing UAV. An on-board autopilot hardware has been designed and built for this UAV. Fig. 8 shows a functional diagram of the autopilot.

Two ARM7 microcontrollers are used for sensor data handling and navigation algorithms. Both are connected with a UART channel. Flight data are measured by different sensors: a GPS receiver, an IMU with magnetometers, and four pressure sensors. One of the pressure sensors is used as a barometer for altitude measurement and the rest, one per axis, are connected to Pitot tubes for air-speed measurement. Sensor data are stored in an SD card for experiment analysis. The IMU used is the ADIS16405 from Analog Devices.

The navigation microcontroller receives processed sensor data from the other microcontroller and sends this data over the radio link. In the future, when the control loop is closed, this microcontroller will send the PWM signals to the actuators. Fig. 9 shows a picture of the autopilot.

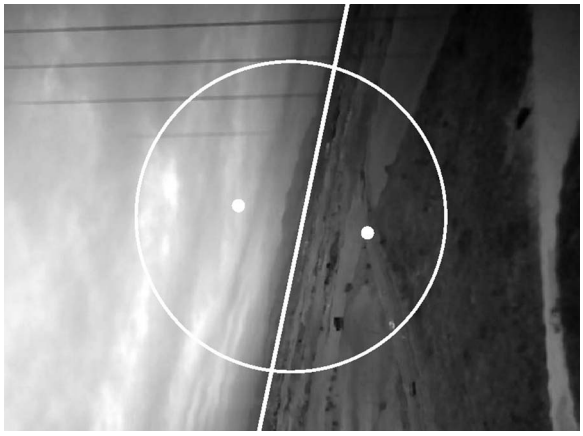

Fig. 10. One of the frames processed by the vision system.

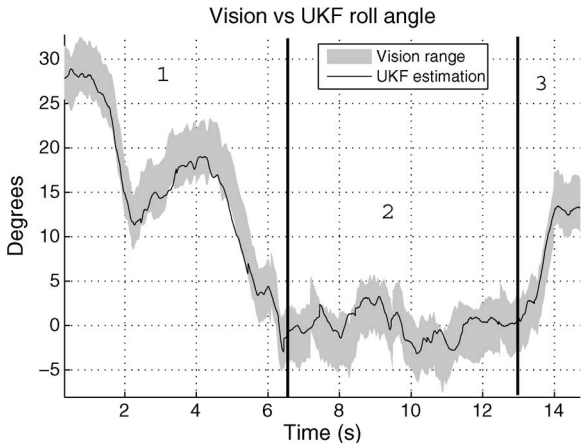

Fig. 11. Roll angle comparison between vision system and UKF estimation.

The target of the experiments is to assess the accuracy of the estimation of each of the three attitude angles. Therefore, experiments are designed to excite the *single modes* of the system, that is to say, pure roll, pure pitch, and pure yaw. However, due to the nature of the system, it is not possible to do so in practice; there is always some coupling between motions. We tried to excite these modes during the experiments. In addition, typical maneuvers such as coordinated turns were also included. A ground station was built to receive data from the UAV using a radio link. The flight is manually controlled using a conventional radio control unit.

For validation purposes, data coming from independent sensors (not used by the estimation algorithm) have been considered. For the roll angle, a computer vision system is used. The GPS velocity is used to validate the yaw angle.

The vision system uses a small camera attached to the UAV. An algorithm was developed to obtain the roll angle from the video measuring the slope of the horizon. The algorithm is based on [30]–[32]. Fig. 10 shows one of the frames taken during the flight and processed by the vision system. The results of this system have an uncertainty of $\pm 3^{\circ}$.

Fig. 11 shows a comparison between the estimated roll angle and the measurements from the vision system. The area marked with "1" depicts the end of a turn to the right. It is followed by a steady flight in the area marked with "2", finally leading to

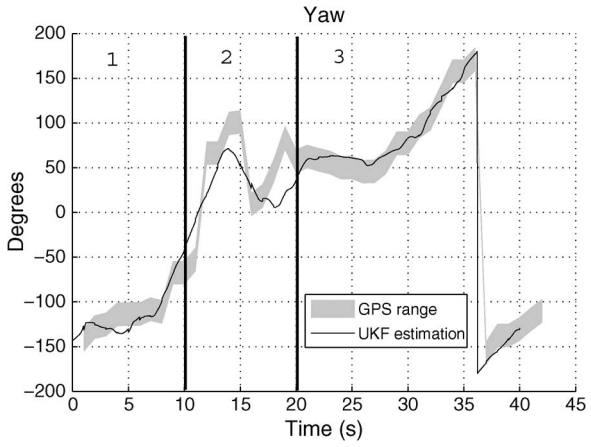

Fig. 12. Yaw angle comparison between GPS system and UKF estimation.

another turn to the right in the area marked with "3". The results of the comparison are clearly satisfactory.

The actual experiment took place under adverse atmospheric conditions, in particular, there were gusts of wind. Hence, when there is strong crosswind, the yaw angle differs from the heading of the plane. The heading is measured using the GPS velocity with an uncertainty of $\pm 10^{\circ}$. It can be noted that the GPS velocity is only used to subtract the centrifugal contribution of the accelerometers [see (46)] in the estimation algorithm. Therefore, it can be taken as an independent system for validation purposes.

Fig. 12 shows a comparison of the estimation of the yaw angle and the heading measured by the GPS. The figure corresponds to a turn of 360°. During the area marked with "1", the UAV faces the wind gusts. The area marked with "2" shows the effect of crosswind; it makes the yaw angle and the heading diverge. In other words, the UAV does not move in the direction it points to. Finally, throughout the area marked with "3", the UAV has tail wind and the yaw angle closely follows the GPS heading.

With the sensors we are currently using, it is not possible to have an independent measurement of the pitch angle. However, due to the formulation of the algorithm using quaternions, the pitch angle is strongly coupled to the roll and yaw angles. Therefore, it can be assumed that if both the roll and yaw angles are correctly estimated, the pitch angle is correctly estimated too.

VII. Conclusion

This paper has considered the problem of attitude estimation of an UAV to establish closed-loop control. The mathematical formulation of the problem has been presented in terms of quaternions. The on-board AHRS is based on a MEMS IMU.

A widely used estimator is the Kalman filter. The kinematic model of aircraft attitude is highly nonlinear, so a version of the Kalman filter able to cope with nonlinearities is needed. Two of these versions have been studied: the EKF and the UKF. A common solution in the satellite attitude estimation practice is the TRIAD algorithm; in this paper, it has been used as the observation model in the UKF framework.

Previous studies in the attitude determination field give the same confidence to the sensors throughout the whole UAV mission. This is unsuitable for missions that mix both acrobatic and non-acrobatic maneuvers. Using the TRIAD algorithm, it is easy to select the most reliable sensors along the different phases of a flight. Adaptive criteria for this selection have been introduced.

A simulation framework based on XPlane 9 has been introduced. This simulation corresponds to the characteristics of the experimental UAV. Using simulation results, it was found that the UKF shows better performance than the EKF. It is shown that the UKF withstands noise much better than the EKF. Therefore, the UKF has been used in the final version of the AHRS.

The performance of the algorithm has been assessed through field experiments. Using independent sensors, it has been checked that the estimation algorithm gives good results. A possible idea for future research is to include the information from these sensors into the AHRS itself. Since the estimation results are encouraging, the next experimental work will be to feed the estimations to a closed-loop controller.

Due to space constraints, only some snapshots of the experimental results have been shown. However, the complete set of data confirm the good quality of the estimations. This gives confidence on the algorithm, which is easy to implement and it can be run on an on-board microcontroller.

References

[1] I. Maza, F. Caballero, J. Capitan, J. M. de Dios, and A. Ollero, "Firemen monitoring with multiple UAVs for search and rescue missions," in *Proc. IEEE Workshop SSRR*, 2010, pp. 1–6.

[2] D. T. Cole, S. Sukkaireh, and A. H. Göktogan, "System development and demonstration of a UAV control architecture for information gathering missions," *J. Field Robot.*, vol. 23, no. 6/7, pp. 417–440, Jun./Jul. 2006.

[3] Y. Xia, Z. Zhu, M. Fu, and S. Wang, "Attitude tracking of rigid spacecraft with bounded disturbances," *IEEE Trans. Ind. Electron.*, vol. 58, no. 2, pp. 647–659, Feb. 2011.

[4] B. Zheng and Y. Zhong, "Robust attitude regulation of a 3-dof helicopter benchmark: Theory and experiments," *IEEE Trans. Ind. Electron.*, vol. 58, no. 2, pp. 660–670, Feb. 2011.

[5] Z. Zhu, Y. Xia, and M. Fu, "Adaptive sliding mode control for attitude stabilization with actuator saturation," *IEEE Trans. Ind. Electron.*, vol. 58, no. 10, pp. 4898–4907, Jan. 2011.

[6] G. Cai, B. Chen, K. Peng, M. Dong, and T. Lee, "Modeling and control of the yaw channel of a UAV helicopter," *IEEE Trans. Ind. Electron.*, vol. 55, no. 9, pp. 3426–3434, Sep. 2008.

[7] O. Špinka, O. Holub, and Z. Hanzálek, "Low-cost reconfigurable control system for small UAVs," *IEEE Trans. Ind. Electron.*, vol. 58, no. 3, pp. 880–889, Mar. 2011.

[8] S. A. Whitmore, M. Fife, and L. Brashear, "Development of a closed-loop strap down attitude system for an ultrahigh altitude flight experiment," NASA, Washington, DC, NASA Tech. Memo. 4775, 1997.

[9] S. J. Julier and J. K. Uhlmann, "A new extension of the Kalman filter to nonlinear systems," in *Proc. Int. Symp. Aerosp./Defense Sens., Simul. Controls*, Orlando, FL, 1997, pp. 182–193.

[10] E. Wan and R. van der Merwe, "The unscencted Kalman filter," in *Kalman Filtering and Neural Networks*, S. Haykin, Ed. Hoboken, NJ: Wiley, 2001, ch. 7.

[11] S. J. Julier and J. K. Uhlmann, "Unscented filtering and nonlinear estimation," *Proc. IEEE*, vol. 92, no. 3, pp. 401–422, Mar. 2004.

[12] A. H. Jazwinski, *Stochastic Processes and Filtering Theory*. New York: Academic, 1970.

[13] K. Reif, S. Gunther, E. Yaz, and R. Unbehauen, "Stochastic stability of the discrete-time extended Kalman filter," *IEEE Trans. Autom. Control*, vol. 44, no. 4, pp. 714–728, Apr. 1999.

[14] 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.

[15] S. P. Won, W. W. Melek, and F. Golnaraghi, "A Kalman/particle filter-based position and orientation estimation method using a position sensor/inertial measurement unit hybrid system," *IEEE Trans. Ind. Electron.*, vol. 57, no. 5, pp. 1787–1798, May 2010.
[16] B. O. Teixeira, L. A. Tôrres, P. Iscold, and L. A. Aguirre, "Flight path reconstruction—A comparison of nonlinear Kalman filter and smoother algorithms," *Aerosp. Sci. Technol.*, vol. 15, no. 1, pp. 60–71, Jan. 2011.
[17] J. Marins, X. Yun, E. Bachmann, R. McGhee, and M. Zyda, "An extended Kalman filter for quaternion-based orientation estimation using marg sensors," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2001, vol. 4, pp. 2003–2011.
[18] M. J. Hale, P. Vergez, M. J. Meerman, and Y. Hashida, "Kalman filtering and the attitude determination and control task," presented at the AIAA Space Conf. Exhibit, San Diego, CA, Sep., 2004, Paper AIAA-2004-6018.
[19] J. Qi, J. Han, and Z. Wu, "Rotorcraft UAV actuator failure estimation with KF-based adaptive UKF algorithm," in *Proc. Amer. Control Conf.*, Jun. 2008, pp. 1618–1623.
[20] A. M. Hasan, K. Samsudin, A. R. Ramli, R. S. Azmir, and S. A. Ismaeel, "A review of navigation systems (integration and algorithms)," *Aust. J. Basic Appl. Sci.*, vol. 3, no. 2, pp. 943–959, 2009.
[21] M. D. Shuster and S. D. Oh, "Three axis attitude determination from vector observations," *J. Guid. Control*, vol. 4, no. 1, pp. 70–77, Jan. 1981.
[22] S. P. Won, F. Golnaraghi, and W. W. Melek, "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.
[23] D. Du, L. Liu, and X. Du, "A low-cost attitude estimation system for UAV application," in *Proc. CCDC*, May 2010, pp. 4489–4492.
[24] S. Lei, H. Chang-qiang, W. Xing-wei, and W. Wen-chao, "Research on a new eight-accelerometer configuration for attitude angle calculation of UAV," in *Proc. CCDC*, May 2010, pp. 4119–4123.
[25] W. YongLiang, W. TianMiao, L. JianHong, W. ChaoLei, and Z. Chen, "Attitude estimation for small helicopter using extended Kalman filter," in *Proc. IEEE Conf. Robot., Autom. Mechatronics*, Sep. 2008, pp. 577–581.
[26] J. S. Jang and D. Liccardo, "Small UAV automation using MEMS," *IEEE Aerosp. Electron. Syst. Mag.*, vol. 22, no. 5, pp. 30–34, May 2007.
[27] D. Gebre-Egziabher, G. Elkaim, J. Powell, and B. Parkinson, "A gyro-free quaternion-based attitude determination system suitable for implementation using low cost sensors," in *Proc. IEEE Position Location Navig. Symp.*, 2000, pp. 185–192.
[28] A. E. Hadri and A. Benallegue, "Sliding mode observer to estimate both the attitude and the gyro-bias by using low-cost sensors," in *Proc. IEEE/RSJ Int. Conf. IROS*, Oct. 2009, pp. 2867–2872.
[29] J. Vaganay, M. Aldon, and A. Fournier, "Mobile robot attitude estimation by fusion of inertial data," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 1993, vol. 1, pp. 277–282.
[30] J.-C. Bazin, I. Kweon, C. Demonceaux, and P. Vasseur, "UAV attitude estimation by vanishing points in catadioptric images," in *Proc. IEEE ICRA*, May 2008, pp. 2743–2749.
[31] H.-Z. Yuan, X.-Q. Zhang, and Z.-L. Feng, "Horizon detection in foggy aerial image," in *Proc. Int. Conf. IASP*, Apr. 2010, pp. 191–194.
[32] G. Pereira, P. Iscold, and L. Torres, "Airplane attitude estimation using computer vision: Simple method and actual experiments," *Electron. Lett.*, vol. 44, no. 22, pp. 1303–1304, Oct. 2008.

Hector Garcia de Marina was born in Madrid. He received the B.Eng. degree in physics from the Complutense University of Madrid, Madrid, Spain, in 2006 and the M.Sc. degree in electronics engineering from the University of Alcalá de Henares (UAH), Alcalá de Henares, Spain, in 2011.

His research interests include real-time simulation, sensor data fusion, and autonomous navigation systems for spacecraft and UAV missions.

Fernando J. Pereda was born in Madrid. He received the B.Eng. degree in computer science and software engineering from the Carlos III University of Madrid, Madrid, Spain, in 2010 and the M.Sc. in computer science from the Complutense University of Madrid, Madrid, in 2011.

His research interests include real-time simulation, sensor data fusion, autonomous navigation systems, and marine craft estimation and control.

Jose M. Giron-Sierra (M'94) was born in Valladolid, Spain. He received the Licentiate and Ph.D. degrees in physics from the Complutense University of Madrid (UCM), Madrid, Spain, in 1972 and 1978, respectively.

He is Full Professor with the Department of Computer Architecture and Automatic Control of the Complutense UCM, from 1988. He has been the author of 180 publications in conference proceedings and journals. He holds two patents for a robot and a communication system. His research is related to applied automatic control and simulation: ships, airplanes and spacecrafts, robotics, logistics, process control.

Dr. Giron-Sierra is a member of the IFAC Technical Committee on Marine Systems; he belongs to IEEE, AIAA, and EUROSIM.

Felipe Espinosa (M'98) received the M.Eng. degree from the Polytechnic University of Madrid (UPM), Madrid, Spain, in 1991, and the Ph.D. degree in telecommunication from the University of Alcala (UAH), Alcalá de Henares, Spain, in 1999.

He became an Associate Professor, in 2000, with the Electronics Department of the University of Alcala, regularly involved in electronic control and automation subjects in several Post-Degree programmes. His current research interests include control, communication and sensorial systems applied to intelligent transportation systems and industrial automation.