

Evaluating the Performance of Euler and Quaternion-Based AHRS Models in Embedded Systems for Aviation and Autonomous Vehicle Applications

Tarık Ünler^{1*} , Yavuz Selim Güler² 

¹Necmettin Erbakan University, Avionics Department, 42100, Meram, Konya, Türkiye. (tunler@erbakan.edu.tr)

²Necmettin Erbakan University, Aerospace Engineering Department, 42100, Meram, Konya, Türkiye. (vzguler@gmail.com)

Article Info

Received: 08 February 2025
Revised: 03 April 2025
Accepted: 29 April 2025
Published Online: 25 June 2025

Keywords:

Kalman Filter
Embedded Systems
Computational Efficiency
Sensor Fusion
IMU
AHRS

Corresponding Author: Tarık Ünler

RESEARCH ARTICLE

<https://doi.org/10.30518/jav.1633060>

Abstract

This study investigates the impact of different Kalman Filter models on the performance of the AHRS system and evaluates its microprocessor-independent computation speed, with particular emphasis on its critical role in aviation and autonomous vehicle applications. AHRS is vital for aircraft stability, navigation, and control by providing accurate attitude estimation. The research employed an MPU-9255 sensor and an ATmega2560 microprocessor, processing data from the sensor to implement Kalman Filters using different mathematical models. Two models, based on Euler angles and quaternions, were tested and compared in terms of measurement accuracy and execution speed. The computation time difference between the models was found to be 10 millisecond (ms). By assessing the performance of these models within an embedded system, the study introduces a novel framework that serves as a reference for optimizing AHRS applications in aviation and other real-time orientation tracking systems.

1. Introduction

The term orientation refers to the position of any object relative to a reference point. Real-time orientation tracking is used in many areas of our lives. Especially since aerial vehicles are now utilized in numerous aspects of our lives (Konar et al. ,2024), this term is frequently used in the field of aviation, particularly in relation to autonomous vehicles. The most well-known examples are autonomous vehicles (Kim & Golnaraghi, 2004). Autonomous vehicles do not require a user inside and can be guided by artificial intelligence or remotely. The orientation of an autonomous vehicle is often referred to as “attitude” (Munguia & Grau, 2011). The attitude information of an autonomous vehicle is calculated using an algorithm called Attitude and Heading Reference Systems (AHRS) using data from accelerometer, gyroscope and magnetometer sensors (Diaz, Müller, Jiménez, & Zampella, 2015). Sensors containing these three (accelerometer, gyroscope and magnetometer) or two (accelerometer, gyroscope) measuring instruments are called Inertial Measurement Unit (IMU). IMU sensors are mostly used with the Kalman Filter (KF) for AHRS. The Kalman Filter is a recursive filter consisting of a set of mathematical equations that allows the position of vehicles to be calculated efficiently (Welch, 2020). For Kalman filter-based AHRS systems, models with different mathematical interpretations

such as Euler angles and quaternions have been developed (Shuster, 1993). In this study, the calculation speed and measurement values of different mathematical models were compared.

The data obtained from IMU sensors are measurements that can be used to calculate the movement or position of the object at certain time intervals. Their low cost and versatility make them a good choice for many applications (Saraf, Moon, & Madotto, 2023). IMUs are often used in combination with a microcontroller (Ferdinando, Khoswanto, & Purwanto, 2012; Nagui, Attallah, Zaghoul, & Morsi, 2020; Vignahala, Ramesh, Devanaboyina, & Reddy, 2021). Microcontrollers have been used in many applications other than IMUs because of their compact size and low power requirements (Samiullah, Irfan, & Rafique).

Microcontroller selection has been an important issue for system designers and a subject that needs to be decided carefully (Parai, Das, Das, & Engineering, 2013). In making this choice, designers consider, among other things, how long it takes the microcontroller to perform the planned action (Gelsinger, 2001). In real-time position tracking, fast calculations are required. Any slow calculation may cause situations such as accidents and loss of control. The possibility of loss of control and accidents cannot be ignored for vehicles used in critical areas such as unmanned aerial vehicles (UAVs). The areas of use of UAVs include border

security, search and rescue, wildlife research, firefighting, precision agriculture, surveying and mapping (Couturier & Akhloufi, 2021). It also covers critical areas such as the military. In addition, the complexity, cost and required qualifications of projects are increasing day by day (Menghal & Laxmi, 2010). This trend also affects UAV projects.

In this study, among the characteristics of microcontrollers in projects, the computation speed is emphasized. Limited computing capabilities of microcontrollers (Immonen & Hämäläinen, 2022) are a

known problem. Although it is known that the selection of more advanced microcontrollers in the selection of microcontrollers in projects can be a solution to this problem, it is predicted that models with simple mathematical models will work more performance independent of the microcontroller and will be a solution. When this solution is evaluated for the drones given in Table 1, it is seen that only changing the model used will have a positive effect on the product.

Table 1. Drone Types and Qualifications (Emimi, Khaleel, & Alkrash, 2023)

Drones Type	US\$ price	Drawbacks	Advantages	Applications
Rotary Wing (helicopter)	\$20-150k	High price	Hovering, large payload	Supply drops, inspection
Rotary Wing (multicopter)	\$3-50k	Short flight time, small payload	Hovering availability, low price	Photography, filmography, inspection
Fixed Wing	\$20-150k	Launching, landing High price	Large area coverage, long endurance, high speed	Structural inspection, area survey

There are several known ways to calculate orientation in UAVs. In this study, two different models on the Kalman filter are discussed, namely Euler angles and quaternion-based models.

Euler angles (Kang & Park, 2011) and quaternion-based models (Wang, Zhang, & Sun, 2015) are available in the literature. Euler angles and quaternion-based models have respective advantages and disadvantages. Euler angles have a clear physical interpretation and do not contain unnecessary parameters (Hasan et al., 2018). However, Euler angles have a singularity problem in some angles (Fan, Zhu, & Ren, 2016). Quaternions, on the other hand, do not have a singularity problem but do not have a clear physical interpretation (Hasan et al., 2018).

In this study, the performance of Euler angles and quaternion-based AHRS models, which are frequently used in autonomous control applications, is compared. The performance comparisons of the models are tested on the ATmega2560 embedded system. The advantages and disadvantages of these two models are also mentioned in the study. A comparison of the two models in terms of computation speed is also made.

2. Materials and Methods

2.1. Hardware Used

In this study, Arduino Mega 2560 (ATmega2560) was used as microcontroller and MPU-9255 10 DOF (Degrees of Freedom) sensor was used. The sensor specifications are given in Table 2 and the microcontroller specifications are given in Table 3.

Table 2. MPU-9255 Sensor Specifications

Driver IC	
The MPU-9255 is a sensor that includes a 3-axis accelerometer, a 3-axis gyroscope, and a 3-axis digital compass.	BMP280, a digital barometric pressure sensor
Incorporates a 16-bit ADC. The gyroscope has selectable full-scale ranges of ± 250 , ± 500 , ± 1000 , and $\pm 2000^\circ/\text{sec}$. The accelerometer offers full-scale ranges of $\pm 2g$, $\pm 4g$, $\pm 8g$, and $\pm 16g$. The compass provides a full-scale range of $\pm 4800 \mu\text{T}$	Includes an integrated temperature sensor for compensation of temperature measurements. The pressure measurement range spans from 300 to 1100 hPa, which corresponds to altitudes from +9000m to -500m relative to sea level. The accuracy is ± 0.12 hPa ($\pm 1\text{m}$) within the range of 700 hPa to 900 hPa and temperatures from 25°C to 40°C

Table 3. Arduino Mega 2560 (ATmega2560) Specifications

Feature	Description
Core	8-bit AVR
Performance	16 MHz
Flash Memory	256 KB
RAM	8 KB
Digital I/O Pins	54
ADC Channels	16-channel
PWM Output Pins	15
Serial Communication Ports	4 (UART)
I2C	1 (TWI)
SPI	1
Operating Voltage	5V

Sensor used in this study is MPU-9255 10 Degrees of Freedom (DOF) sensor is given in figure 1.

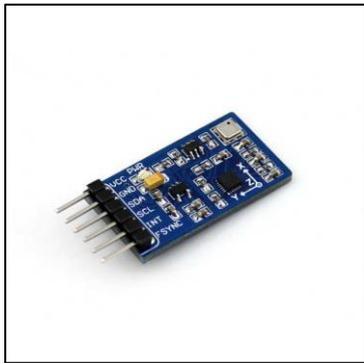


Figure 1. MPU-9255 10 DOF Sensor

It is a frequently used sensor for Inertial Navigation Systems (INS) as a 10 DOF sensor composed of a 3-axis accelerometer, 3-axis gyroscope, 3-axis magnetometer, and one pressure sensor.

The sensor and microcontroller wiring were done as illustrated in Figure 2 and the microcontroller and sensor were powered via USB.

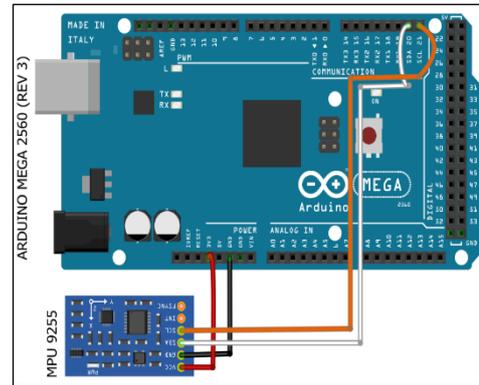


Figure 2. Sensor and Microcontroller Connections

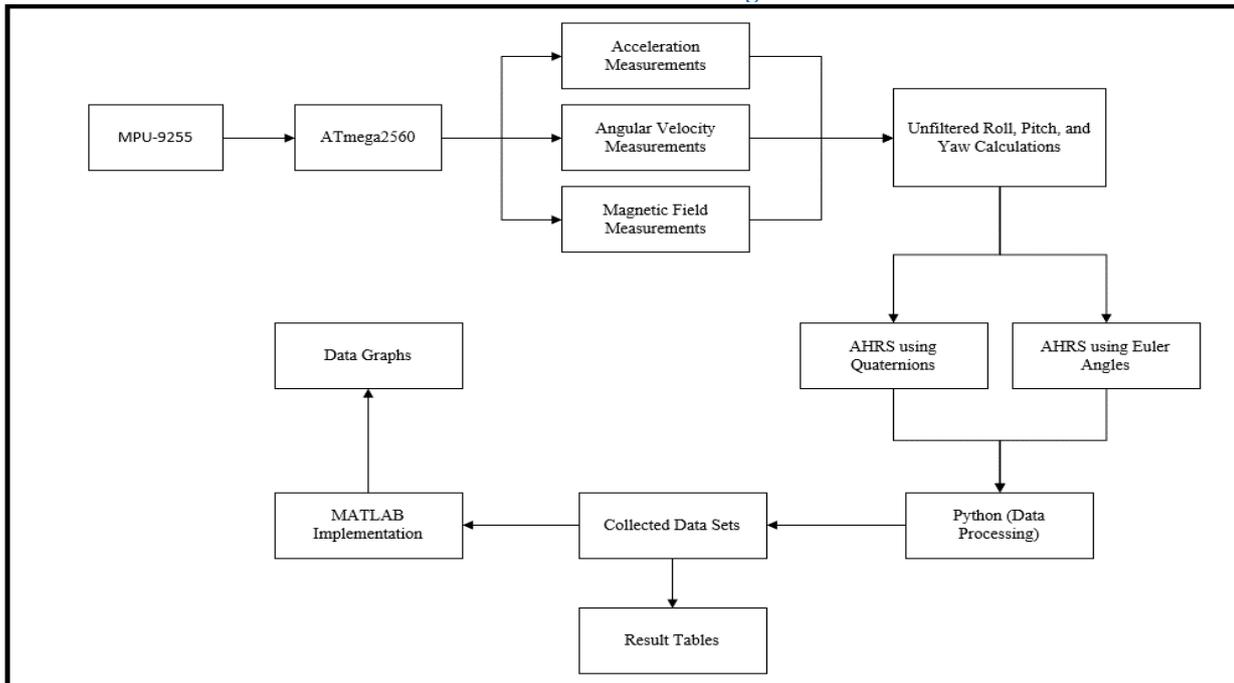


Figure 3. Block Diagram of Experimental System

3. Magnetometer Calibration, Calculation of Euler Angles and Calculation of Quaternions

This section outlines the procedures for magnetometer calibration, as well as the mathematical approaches used to calculate orientation in terms of Euler angles and quaternions. These steps are essential for ensuring accurate and reliable Attitude and Heading Reference System (AHRS) performance.

3.1. Magnetometer Calibration and Gyroscope Measurement Model

A three-axis gyroscope will measure the angular rate about the x, y and z axes of the sensor frame, labeled p, q and r respectively and given in equation 1. The gyroscope values are divided by the scale factor specified in the datasheet before use. The gyroscope measurement model used in this paper is given in equation 2 (Lam, Stamatakos, Woodruff, & Ashton, 2003).

$$\omega = \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{1}$$

$$\omega_{used} = \frac{\omega}{scalefactor} + b + n \tag{2}$$

Where b is the gyroscope bias and n is the white noise that distorts the gyroscope rate measurement.

The magnetometer scale factor is determined according to the sensor data sheet and the offset values are calculated as indicated below. The sensitivity adjustment data for each axis is saved in the sensor ROM during production and is indicated by adding ASA (Asahi Sensitivity Adjustment) to the beginning of the relevant axis

ASAX: Value to be used in magnetometer X axis sensitivity adjustment.

ASAY: Value to be used in magnetometer Y axis sensitivity setting.

ASAZ: Value to be used in magnetometer Z axis sensitivity setting. The equation used to calculate the magnetometer scale factor is given in equation 3.

$$Scale\ Factor_i = \frac{(ASA_i - 128) \times 0.5}{128} + 1 \quad (3)$$

In Equation 1, i represented the x, y or z axis. The calculated scale factors are shown in the study as X_{sf} , Y_{sf} or Z_{sf} for the three axes.

The magnetometer offset values were found using Equations 4, 5 and 6 for the three axes (Poulose, Kim, & Han, 2019).

$$X_{off} = \left(\frac{Mx_{max} - Mx_{min}}{2} \right) - Mx_{max} X_{sf} \quad (4)$$

$$Y_{off} = \left(\frac{My_{max} - My_{min}}{2} \right) - My_{max} Y_{sf} \quad (5)$$

$$Z_{off} = \left(\frac{Mz_{max} - Mz_{min}}{2} \right) - Mz_{max} Z_{sf} \quad (6)$$

In Equations 2, 3 and 4, the offsets in the three axes are denoted by X_{off} , Y_{off} and Z_{off} . Mx , My and Mz represent the raw magnetometer readings for the three axes. The calibrated values of the magnetometer readings for the X, Y and Z axes are calculated in equations 7, 8 and 9 (Poulose et al.,2019).

$$MX = X_{sf} \times M_x + X_{off} \quad (7)$$

$$MY = Y_{sf} \times M_y + Y_{off} \quad (8)$$

$$MZ = Z_{sf} \times M_z + Z_{off} \quad (9)$$

Here MX , MY , MZ are used as calibrated magnetometer data in this study.

3.2. Calculation of Euler Angles

Euler angles were calculated with accelerometer and magnetometer data in equations 10, 11 and 12 (Hanafi, Abozied, Elhalwagy, & Elfarouk, 2019; Tomaszewski, Rapiński, Pelc-Mieczkowska, & Geoinformatics, 2017).

$$Roll(\phi) = \arctan\left(\frac{Ay}{Az}\right) \quad (10)$$

$$Pitch(\theta) = \arctan\left(\frac{-Ax}{\sqrt{Ay^2 + Az^2}}\right) \quad (11)$$

$$Yaw(\varphi) = \arctan\left(\frac{\cos(\phi) MY - \sin(\phi) MZ}{\cos(\theta) MX + \sin(\theta) \sin(\phi) MY + \sin(\theta) \cos(\phi) MZ}\right) \quad (12)$$

A_x , A_y and A_z correspond to accelerometer measurements for all three axes, MX , MY and MZ are calibrated magnetometer measurements along the three axes.

3.3. Calculation of Quaternions

Quaternions are tools used to represent three-dimensional rotations. Quaternions are found in equation 13 (P. Kim & Huh, 2011) using Euler angles.

$$\begin{pmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{pmatrix} = \begin{bmatrix} \cos\frac{\phi}{2} \cos\frac{\theta}{2} \cos\frac{\varphi}{2} + \sin\frac{\phi}{2} \sin\frac{\theta}{2} \sin\frac{\varphi}{2} \\ \sin\frac{\phi}{2} \cos\frac{\theta}{2} \cos\frac{\varphi}{2} - \cos\frac{\phi}{2} \sin\frac{\theta}{2} \sin\frac{\varphi}{2} \\ \cos\frac{\phi}{2} \sin\frac{\theta}{2} \cos\frac{\varphi}{2} + \sin\frac{\phi}{2} \cos\frac{\theta}{2} \sin\frac{\varphi}{2} \\ \cos\frac{\phi}{2} \cos\frac{\theta}{2} \sin\frac{\varphi}{2} + \sin\frac{\phi}{2} \sin\frac{\theta}{2} \cos\frac{\varphi}{2} \end{bmatrix} \quad (13)$$

Here q_1 is the scalar component, q_2 , q_3 and q_4 are the vectorial components of the quaternions.

3.4. Kalman Filter

Kalman Filter based AHRS is used in this paper. The Kalman Filter was created by Rudolf Emil Kalman to solve the filtering problem in aerospace and aircraft (Kalman, 1960). The algorithm of the Kalman Filter is provided in figure 4 (P. Kim & Huh, 2011).

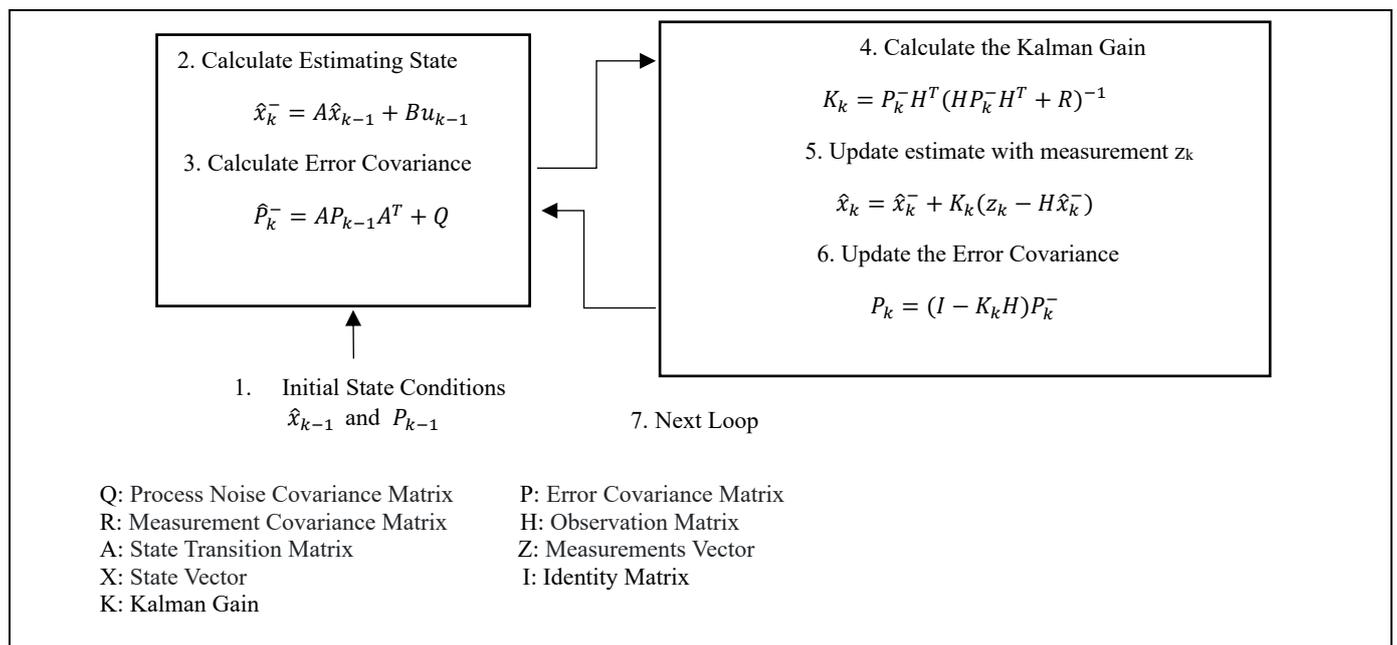


Figure 4. Kalman Filter Algorithm

Step 1 is based on determining the initial conditions of the filter. The error covariance (P), the process noise matrix (Q) and the measurement error matrix (R) are also determined here.

Step 2 is the first step of the estimation part. The measurements at time k-1 are approximated with the state transition matrix and the measurements at time k are approximated with the system control variables.

In step 3, the error covariance matrix is updated, while the Q matrix takes into account biases and uncertainties due to model inaccuracies and integrates them into the error covariance update.

In step 4, the Kalman gain is calculated. The Kalman gain allows to add or subtract the amount of error according to the accuracy of the measurements, which take values between 0-1.

In step 5, the previously calculated (step 1) measurements are updated with the changes due to the error and the new value is obtained with these values in the next iteration.

In step 6 the error covariance matrix is updated again and used to update the new error covariance matrix in step 2.

In step 7 the Kalman filter starts repeating all these steps for the next measurements.

The Kalman Filter (KF) is one of the structures frequently used in AHRS algorithms (Pourtakdoust, Ghanbarpour Asl, & Technology, 2007). AHRS is an algorithm used to define the orientation of a device. KF is one of the filters that allows us to calculate the orientation with data from multiple sensors for AHRS.

3.5. Euler-based AHRS Model

The state vector x in the model consists of roll (ϕ), pitch (θ) and yaw (φ) values and is given as a vector in Equation 14.

$$x = \begin{pmatrix} \phi \\ \theta \\ \varphi \end{pmatrix} \quad (14)$$

The initial error covariance matrix and the process noise covariance matrix of the estimated state are identical. The roll, pitch and yaw covariances are represented on the diagonal respectively and given in equations 15 and 16.

$$P_0 = \begin{bmatrix} \sigma_\phi^2 & 0 & 0 \\ 0 & \sigma_\theta^2 & 0 \\ 0 & 0 & \sigma_\varphi^2 \end{bmatrix} \quad (15)$$

$$Q = \begin{bmatrix} \sigma_\phi^2 & 0 & 0 \\ 0 & \sigma_\theta^2 & 0 \\ 0 & 0 & \sigma_\varphi^2 \end{bmatrix} \quad (16)$$

The sensor measurement error is represented on the diagonal of the measurement covariance matrix and is given in equation 17.

$$R = \begin{bmatrix} \sigma_{sensor\ error}^2 & 0 & 0 \\ 0 & \sigma_{sensor\ error}^2 & 0 \\ 0 & 0 & \sigma_{sensor\ error}^2 \end{bmatrix} \quad (17)$$

The state transfer matrix is used to calculate the next (k) measurements with the previous (k-1) measurements and is given in equation 18 as a 3x3 unit matrix in this model.

$$A = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (18)$$

The control matrix is used to add the effect of the gyro measurements on the system. The control matrix is given in equation 19.

$$B = \begin{bmatrix} dt & 0 & 0 \\ 0 & dt & 0 \\ 0 & 0 & dt \end{bmatrix} \quad (19)$$

'dt' in matrix B is the discrete time. The input vector consists of gyroscope measurements and is given in equation 20.

$$U = \begin{pmatrix} p \\ q \\ r \end{pmatrix} \quad (20)$$

'p, q and r' represent the gyroscope measurements in 3 axes. The measurement pattern vector is used to correct the measurements and is given in equation 21.

$$z_k = \begin{pmatrix} \arctan\left(\frac{Ay}{Az}\right) \\ \arctan\left(\frac{-Ax}{\sqrt{Ay^2 + Az^2}}\right) \\ \arctan\left(\frac{\cos(\theta)MY - \sin(\theta)MZ}{\cos(\theta)MX + \sin(\theta)\sin(\theta)MY + \sin(\theta)\cos(\theta)MZ}\right) \end{pmatrix} \quad (21)$$

The measurement transfer matrix H is a 3x3 unit matrix that allows us to correct the measurements by subtracting the measurements obtained in the estimation step from the measurement vector and is given in Equation 22.

$$H = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (22)$$

3.6. Quaternion Based AHRS Model

The state vector x contains the four-quaternion data used to compute the orientation and is given in equation 23.

$$x = \begin{pmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{pmatrix} \quad (23)$$

The relationship between the angular velocity and the rate of change in quaternion is given in equation 24(Wen-shu, Liao-ni, & Qi, 2010).

$$\dot{q} = \frac{1}{2} \begin{bmatrix} 0 & -p & -q & -r \\ p & 0 & r & -q \\ q & -r & 0 & p \\ r & q & -p & 0 \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} = \frac{1}{2} \Omega_k q_{k-1} \quad (24)$$

The dynamic update function representing the quaternions is given in equation 25.

$$\hat{q}_t = f(q_{t-1}, \omega_t) = \left(I_4 + \frac{dt}{2} \Omega_k \right) q_{k-1} \quad (25)$$

In the model used in this study, the state transfer matrix is a vector and nonlinear. It is linearized by calculating the Jacobian of the transfer matrix. The nonlinear transfer matrix

is given in equation 26.

$$f(q, \omega) = \begin{bmatrix} q_1 - \frac{dt}{2}pq_2 - \frac{dt}{2}qq_3 - \frac{dt}{2}rq_4 \\ q_2 + \frac{dt}{2}pq_1 - \frac{dt}{2}qq_4 + \frac{dt}{2}rq_3 \\ q_3 + \frac{dt}{2}pq_z + \frac{dt}{2}qq_1 - \frac{dt}{2}rq_2 \\ q_4 - \frac{dt}{2}pq_3 + \frac{dt}{2}qq_2 + \frac{dt}{2}rq_1 \end{bmatrix} \quad (26)$$

Equations 27 and 28 give the linearization operations by taking the Jacobian matrix. The linearized transfer matrix is given in equation 29.

$$\nabla f(q_{t-1}, \omega_t) = \frac{\partial f(q_{t-1}, \omega_t)}{\partial q} \quad (27)$$

$$= \begin{bmatrix} \frac{\partial f(q_{t-1}, \omega_t)}{\partial q_1} & \frac{\partial f(q_{t-1}, \omega_t)}{\partial q_2} & \frac{\partial f(q_{t-1}, \omega_t)}{\partial q_3} & \frac{\partial f(q_{t-1}, \omega_t)}{\partial q_4} \end{bmatrix} \quad (28)$$

$$\nabla f = \begin{bmatrix} 1 & -\frac{dt}{2}p & -\frac{dt}{2}q & -\frac{dt}{2}r \\ \frac{dt}{2}p & 1 & \frac{dt}{2}r & -\frac{dt}{2}q \\ \frac{dt}{2}q & -\frac{dt}{2}r & 1 & \frac{dt}{2}p \\ \frac{dt}{2}r & \frac{dt}{2}q & -\frac{dt}{2}p & 1 \end{bmatrix} = A \quad (29)$$

The measurement model vector will help us to correct the measurements and is given in equation 30.

$$z_k = \begin{bmatrix} \cos \frac{\phi}{2} \cos \frac{\theta}{2} \cos \frac{\varphi}{2} + \sin \frac{\phi}{2} \sin \frac{\theta}{2} \sin \frac{\varphi}{2} \\ \sin \frac{\phi}{2} \cos \frac{\theta}{2} \cos \frac{\varphi}{2} - \cos \frac{\phi}{2} \sin \frac{\theta}{2} \sin \frac{\varphi}{2} \\ \cos \frac{\phi}{2} \sin \frac{\theta}{2} \cos \frac{\varphi}{2} + \sin \frac{\phi}{2} \cos \frac{\theta}{2} \sin \frac{\varphi}{2} \\ \cos \frac{\phi}{2} \cos \frac{\theta}{2} \sin \frac{\varphi}{2} + \sin \frac{\phi}{2} \sin \frac{\theta}{2} \cos \frac{\varphi}{2} \end{bmatrix} \quad (30)$$

The measurement transfer matrix H is a 4x4 unit matrix that allows us to correct the measurements by subtracting the measurements obtained in the estimation step from the measurement vector and is given in equation 31.

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (31)$$

The AHRS models, whose mathematical models are given above, were operated and compared in the experimental setup in Figure 5.

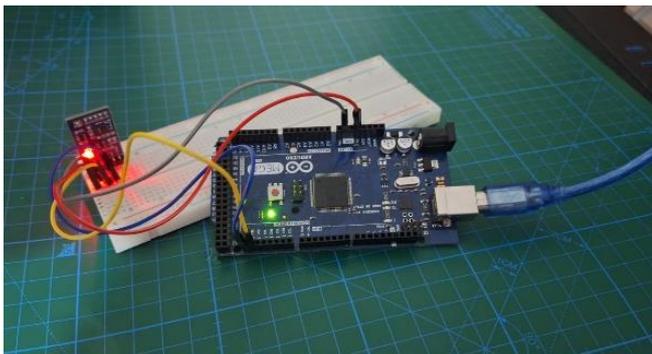


Figure 5. Experimental Setup

4. Results and Discussion

In this study, two different AHRS models were compared on the embedded system under the same conditions. Before the comparison, the model outputs at specific angles were recorded. Graphs and tables below show the readings from the two distinct models used in the study at identical angles. To ensure methodological alignment between graphical and tabular representations, both formats were prepared using the same 15 measurements, which were considered sufficient for an accurate comparison of model performance.

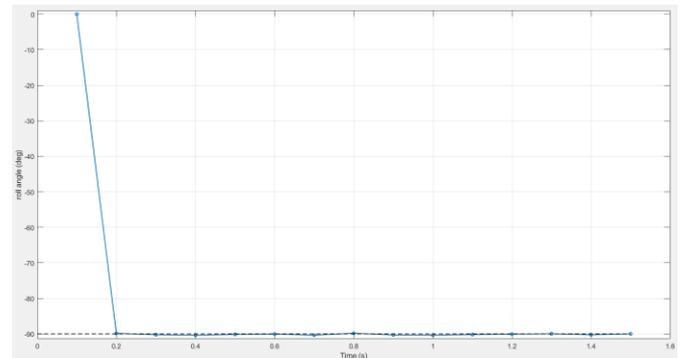


Figure 6. Roll at -90-degree Euler Model

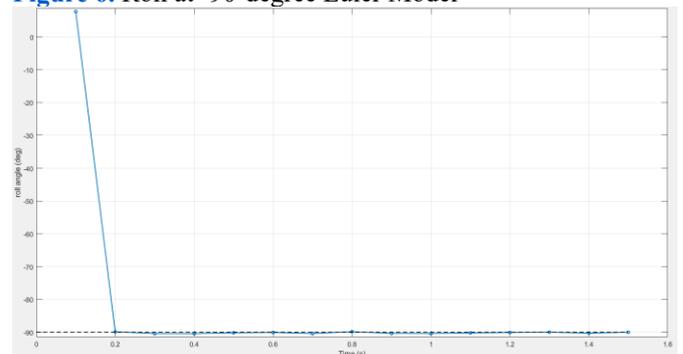


Figure 7. Roll at -90-degree Quaternion Model

Figures 6 and 7 illustrate the results of two different AHRS models for roll angle at -90 degrees. Table 4 presents the values of the two different AHRS models at -90 degrees.

Table 4. Roll Angle Values at -90 Degrees

Reading Order	Quaternion Based AHRS	Euler Angles Based Model AHRS	Time(s)
1	7.73	0.04	0.1
2	-89.85	-89.86	0.2
3	-90.40	-90.26	0.3
4	-90.42	-90.40	0.4
5	-90.18	-90.18	0.5
6	-90.07	-90.06	0.6
7	-90.40	-90.38	0.7
8	-89.82	-89.83	0.8
9	-90.35	-90.32	0.9
10	-90.39	-90.38	1
11	-90.22	-90.22	1.1
12	-90.08	-90.08	1.2
13	-89.98	-89.98	1.3
14	-90.29	-90.27	1.4
15	-90.00	-90.00	1.5

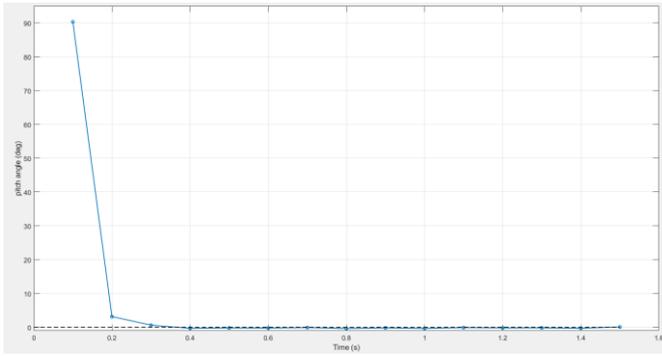


Figure 8. Pitch degree at 0-degree Euler Model

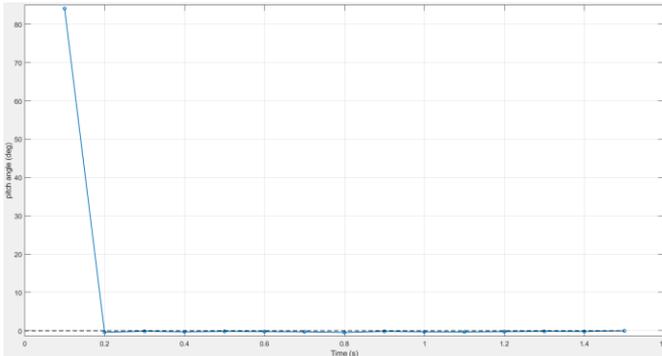


Figure 9. Pitch degree at 0-degree Quaternion Model

Figures 8 and 9 illustrate the results of two different AHRS models for pitch angle at 0 degrees. Table 5 shows the values of two different AHRS models at 0 degrees.

Table 5. 0 Degree Pitch Angle Values

Reading Order	Quaternion Based AHRS	Euler Angles Based Model AHRS	Time(s)
1	84.10	90.29	0.1
2	-0.40	3.16	0.2
3	-0.14	0.63	0.3
4	-0.29	-0.30	0.4
5	-0.17	-0.19	0.5
6	-0.23	-0.23	0.6
7	-0.28	-0.09	0.7
8	-0.46	-0.35	0.8
9	-0.17	-0.18	0.9
10	-0.29	-0.33	1
11	-0.32	-0.10	1.1
12	-0.24	-0.18	1.2
13	-0.16	-0.17	1.3
14	-0.20	-0.29	1.4
15	-0.02	0.10	1.5

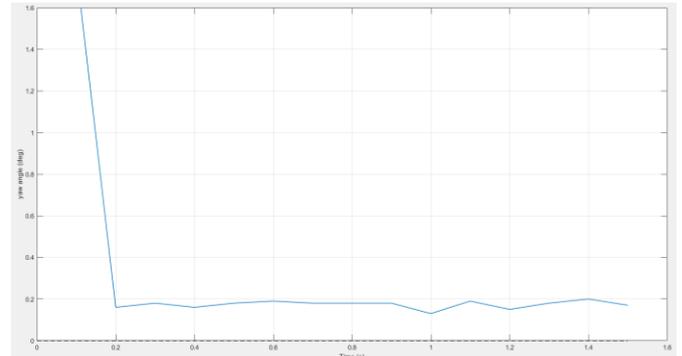


Figure 10. Yaw at 0-degree Quaternion Model

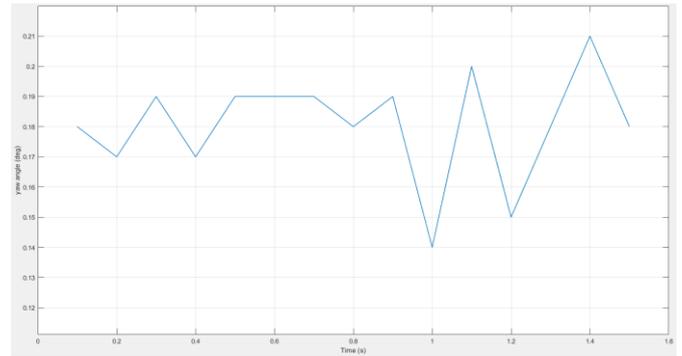


Figure 11. Yaw at 0-degree Euler Model

Figures 10 and 11 illustrate the results of two different AHRS models for a yaw angle at 0 degrees. Table 6 shows the values of two different AHRS models at 0 degrees.

Table 6. Yaw Angle Values at 0 Degree

Reading Order	Quaternion Based AHRS	Euler Angles Based Model AHRS	Time(s)
1	1.79	0.18	0.1
2	0.16	0.17	0.2
3	0.18	0.19	0.3
4	0.16	0.17	0.4
5	0.18	0.19	0.5
6	0.19	0.19	0.6
7	0.18	0.19	0.7
8	0.18	0.18	0.8
9	0.18	0.19	0.9
10	0.13	0.14	1
11	0.19	0.20	1.1
12	0.15	0.15	1.2
13	0.18	0.18	1.3
14	0.20	0.21	1.4
15	0.17	0.18	1.5

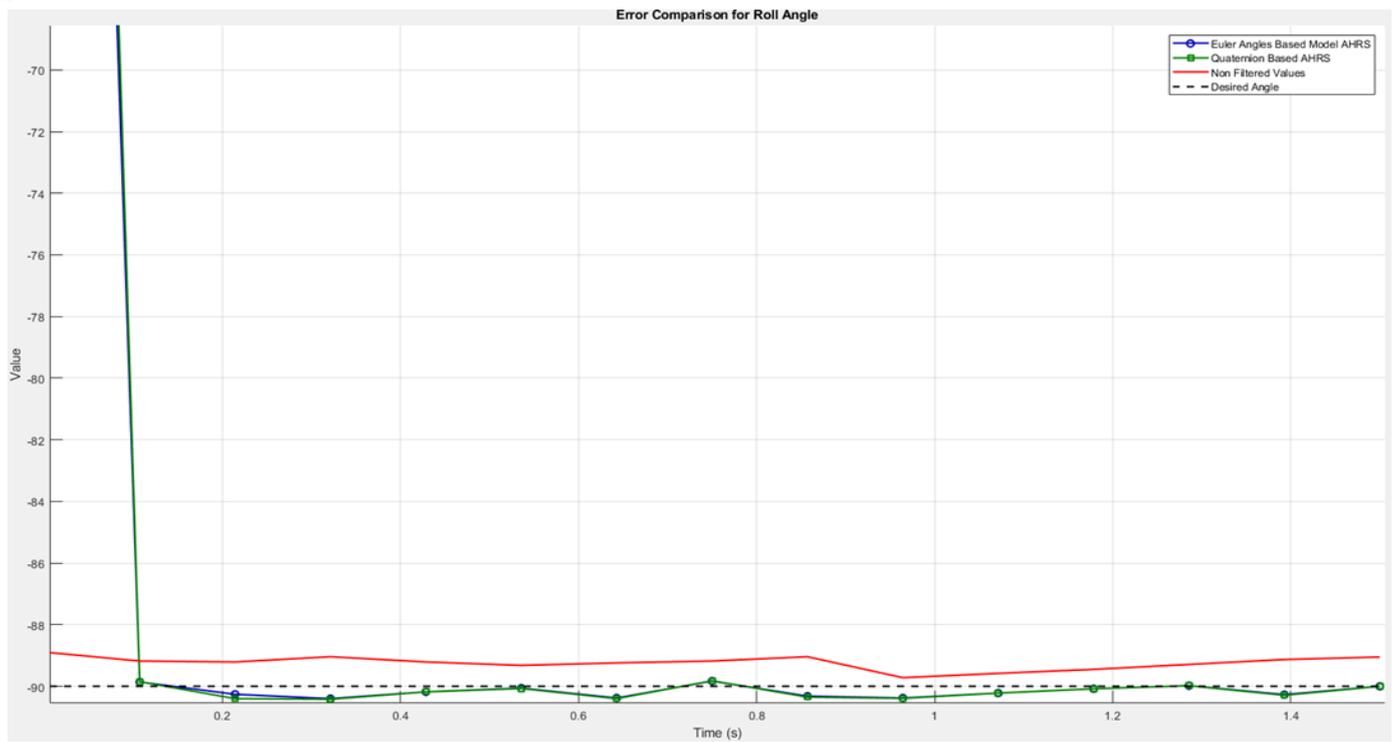


Figure 12. Roll Angle Comparison

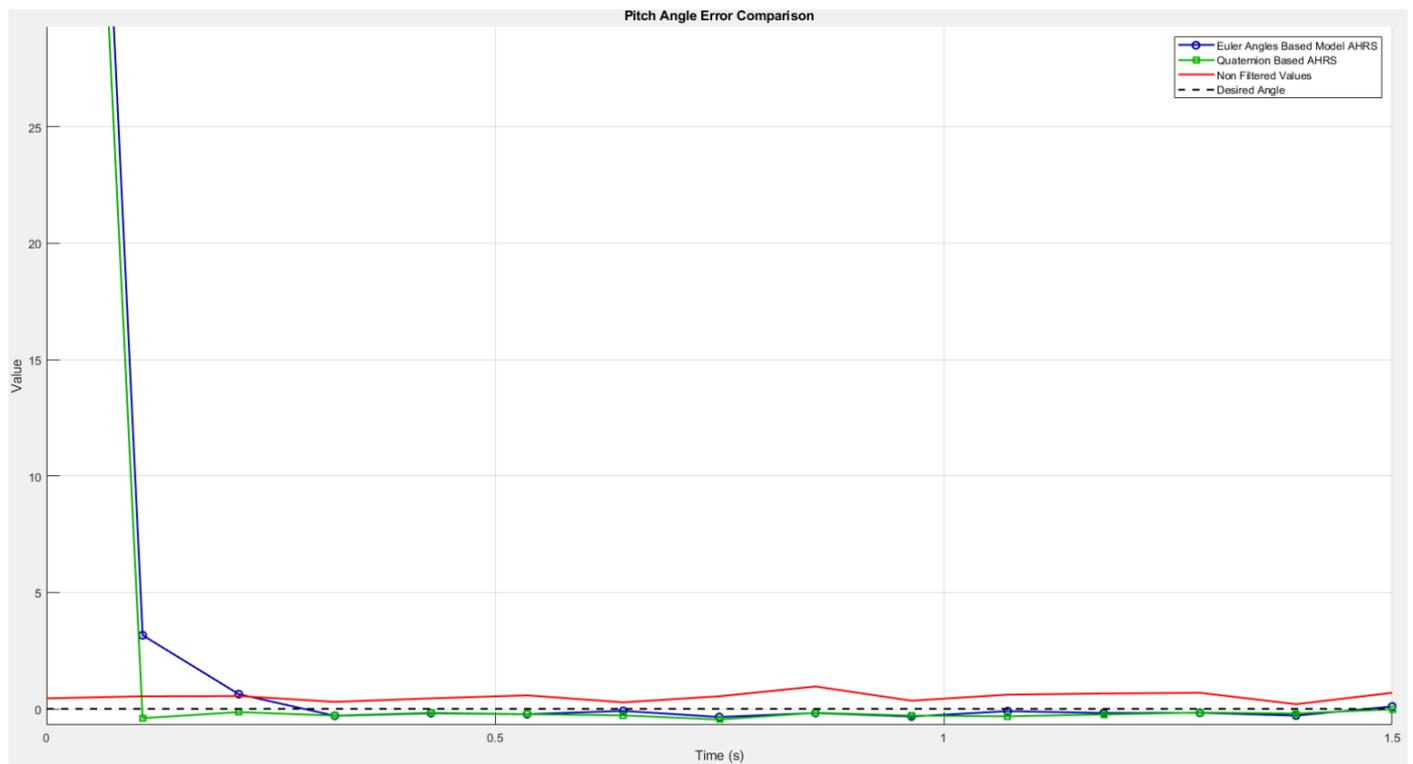


Figure 13. Pitch Angle Comparison

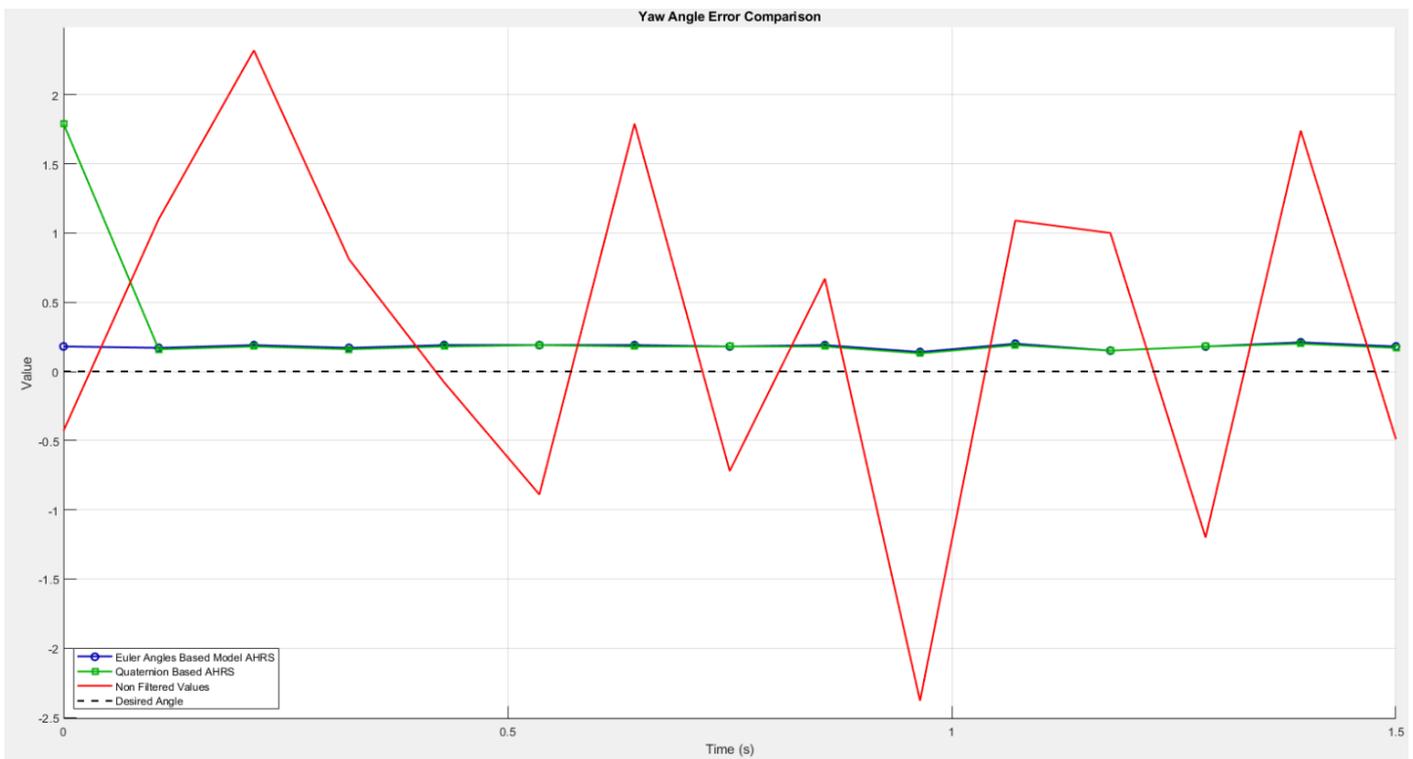


Figure 14. Yaw Angle Comparison

According to Tables 4, 5 and 6, it is observed that two different models give close values for the roll angle in terms of experimental values to the expected theoretical value in measurements made in the same environment and conditions. This shows that both models can calculate the roll angle with a certain accuracy. However, the results of the models do not completely coincide with the theoretical value and contain some margin of error.

For the pitch angle, significant differences were observed between iterations, especially at the fifteenth iteration, where the quaternion-based model gave the closest value to the theoretical value. These findings show that the quaternion-based model shows higher accuracy at certain iteration numbers. Nonetheless, it is essential to recognize that the calculated values have a certain degree of error.

For the yaw angle, although the two models give similar results, it is observed that the precision is high and the accuracy is relatively low when compared to other angles in converging to the theoretical value. The accuracy of the yaw angle is lower than the other angles and there are significant error margins in the calculated results. This finding shows that the models may have certain limitations in the calculation of the yaw angle and are open to improvement.

Figures 12, 13 and 14 illustrate the AHRS models and how far the unfiltered measurements deviate from the true value, i.e., the amount of error they have. According to the graphs, the two different AHRS models are closer to the true value with smaller errors than the unfiltered measurements.

The performances of the Euler and quaternion models were compared using various metrics:

Computation Time: The calculations for the quaternion-based AHRS model, though inherently more complex,

demonstrated comparable performance to the Euler angles-based model when optimized for computation time. Measurement of computation times using the embedded system timer indicated that the quaternion-based model had a time of 15 milliseconds, while the Euler angles-based model achieved a time of 5 milliseconds. The difference in computation times between the two models was quantified as 10 milliseconds.

Memory Utilization: Euler angles-based model used less memory while quaternions required more memory.

Accuracy: It was observed that the quaternion-based model approached the theoretical value more successfully than the Euler angles-based model in advanced iterations.

In addition, the quadrotor control law may affect the results depending on the characteristics of the model used. Euler angles, although a simple and straightforward method, can lead to singularity problems such as the gimbal lock problem, which can adversely affect computational accuracy and system stability. In contrast, quaternions avoid such problems and provide more stable and accurate control. The Kalman filter can be effectively used with both models for noise reduction and state estimation. However, quaternion-based control laws offer higher accuracy and system stability than Euler angles because quaternions perform better without encountering transformation problems like Euler angles. In the literature, it is emphasized that quaternion-based control laws are more robust and accurate than Euler angles (Lei, Liu, & Wang, 2024; Zhi, Li, Song, Yu, & Zhang, 2017).

5. Conclusion

This paper presents a comparative analysis of the Euler and quaternion models for embedded systems in real-time

applications, emphasizing their significance in aviation, where AHRS plays a vital role in ensuring precise attitude estimation for navigation, stability, and control. The study recommends quaternions for applications requiring higher accuracy and stability, while Euler angles are more suitable for simpler, lightweight implementations. Significant findings contribute to enhancing the accuracy and reliability of AHRS models in aviation systems. By adapting mathematical models to an embedded system and conducting performance tests, the study provides valuable insights into algorithm optimization and sensor calibration. The impact of computation time differences on system performance was measured at 10 ms using the experimental setup, with fifteen measurements taken for each angle to compare the models. While both models produced similar results for the roll angle, the quaternion-based model demonstrated superior alignment with theoretical values for the pitch angle after multiple iterations, confirming its advantage in long-term accuracy. Additionally, the study underscores the importance of computation speed, particularly for autonomous aircraft and other aviation applications. To further enhance AHRS performance, the paper proposes models that can serve as benchmarks for future tests on different embedded platforms. Moreover, the development or integration of alternative algorithms to improve yaw angle accuracy will further increase the reliability and effectiveness of AHRS in aviation. This research provides a crucial reference for engineers and researchers working on real-time orientation calculations in embedded aviation systems, guiding advancements in flight control and navigation technologies.

Conflicts of Interest

The authors declare that they have no conflicts of interest related to this work.

Author Contributions

The authors declare that they have contributed equally to the article.

References

- Couturier, A., & Akhloufi, M. A. (2021). A review on absolute visual localization for UAV. *Robotics and Autonomous Systems*, 135, 103666.
- Diaz, E. M., Müller, F. d. P., Jiménez, A. R., & Zampella, F. (2015). Evaluation of AHRS algorithms for inertial personal localization in industrial environments. Paper presented at the 2015 IEEE International Conference on Industrial Technology (ICIT).
- Emimi, M., Khaleel, M., & Alkrash, A. (2023). The Current Opportunities and Challenges in Drone Technology. *Int. J. Electr. Eng. and Sustain.*, 1(3), 74-89.
- Fan, W., Zhu, W. D., & Ren, H. (2016). A New Singularity-Free Formulation of a Three-Dimensional Euler-Bernoulli Beam Using Euler Parameters. *Journal of Computational and Nonlinear Dynamics*, 11(4).
- Ferdinando, H., Khoswanto, H., & Purwanto, D. (2012, 2-4 July 2012). Embedded Kalman Filter for Inertial Measurement Unit (IMU) on the ATmega8535. Paper presented at the 2012 International Symposium on Innovations in Intelligent Systems and Applications.
- Gelsinger, P. P. (2001). Microprocessors for the new millennium: Challenges, opportunities, and new frontiers. Paper presented at the 2001 IEEE International Solid-State Circuits Conference. Digest of Technical Papers. ISSCC (Cat. No.01CH37177).
- Hanafi, M. E., Abozied, M. A., Elhalwagy, Y. Z., & Elfarouk, A. O. (2019). Multi-sensory data fusion for high performance attitude estimation. Paper presented at the International Conference on Aerospace Sciences and Aviation Technology.
- Hasan, M., Abdullah, A., Ahmed, A., Hamzah, N., Said, M., Yaacob, S., Engineering, C. (2018). Analysis on Euler angles rotation of a rigid body in three-axis attitude based on RazakSAT data. *10(1-14)*, 73-76.
- Immonen, R., & Hämäläinen, T. (2022). Tiny Machine Learning for Resource-Constrained Microcontrollers. *Journal of Sensors*, 2022, 7437023.
- Kalman, R. E. (1960). A New Approach to Linear Filtering and Prediction Problems. *Journal of Basic Engineering*, 82(1), 35-45. doi:10.1115/1.3662552 %J Journal of Basic Engineering
- Kang, C. W., & Park, C. G. (2011). Euler Angle Based Attitude Estimation Avoiding the Singularity Problem. *IFAC Proceedings Volumes*, 44(1), 2096-2102.
- Kim, A., & Golnaraghi, M. (2004). A Quaternion-Based Orientation Estimation Algorithm Using an Inertial Measurement Unit.
- Kim, P., & Huh, L. J. (2011). Kalman filter for beginners: with MATLAB examples.
- Konar, M., Hatipoğlu, S. A., & Akpınar, M. (2024). Improvement of UAV thrust using the BSO algorithm-based ANFIS model. *The Aeronautical Journal*, 1-10.
- Lam, Q., Stamatakos, N., Woodruff, C., & Ashton, S. (2003). Gyro Modeling and Estimation of Its Random Noise Sources (Vol. 5562).
- Lei, B., Liu, B., & Wang, C. (2024). Robust Geometric Control for a Quadrotor UAV with Extended Kalman Filter Estimation. *13(6)*, 205.
- Menghal, P. M., & Laxmi, A. J. (2010). Real time control of electrical machine drives: A review. Paper presented at the 2010 International Conference on Power, Control and Embedded Systems.
- Munguia, R., & Grau, A. (2011). Attitude and Heading System based on EKF total state configuration. Paper presented at the 2011 IEEE International Symposium on Industrial Electronics.
- Nagui, N., Attallah, O., Zaghloul, M. S., & Morsi, I. (2020, 28-31 Oct. 2020). Smart Real-Time Autonomous Navigation System using integration of MEMS-based Low-Cost IMU/GPS. Paper presented at the 2020 11th IEEE Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON).
- Parai, M. K., Das, B., Das, G. J. I. J. o. S. C., & Engineering. (2013). An overview of microcontroller unit: from proper selection to specific application. *2(6)*, 228-231.
- Poulose, A., Kim, J., & Han, D. S. (2019). Indoor Localization with Smartphones: Magnetometer Calibration. Paper presented at the 2019 IEEE International Conference on Consumer Electronics (ICCE).
- Pourtakdoust, S., Ghanbarpour Asl, H. J. A. E., & Technology, A. (2007). An adaptive unscented Kalman filter for quaternion-based orientation estimation in low-cost AHRS. *79(5)*, 485-493.

- Samiullah, M., Irfan, M. Z., & Rafique, A. Microcontrollers: A Comprehensive Overview and Comparative Analysis of Diverse Types.
- Saraf, A., Moon, S., & Madotto, A. (2023, 4-10 June 2023). A Survey of Datasets, Applications, and Models for IMU Sensor Signals. Paper presented at the 2023 IEEE International Conference on Acoustics, Speech, and Signal Processing Workshops (ICASSPW).
- Shuster, M. D. J. N. (1993). A survey of attitude representations. 8(9), 439-517.
- Tomaszewski, D., Rapiński, J., Pelc-Mieczkowska, R. J. R. o. G., & Geoinformatics. (2017). Concept of AHRS algorithm designed for platform independent IMU attitude alignment. 104(1), 33-47.
- Vigrahala, J., Ramesh, N. V. K., Devanaboyina, V. R., & Reddy, B. N. K. (2021, 18-19 June 2021). Attitude, Position and Velocity determination using Low-cost Inertial Measurement Unit for Global Navigation Satellite System Outages. Paper presented at the 2021 10th IEEE International Conference on Communication Systems and Network Technologies (CSNT).
- Wang, L., Zhang, Z., & Sun, P. (2015). Quaternion-Based Kalman Filter for AHRS Using an Adaptive-Step Gradient Descent Algorithm. 12(9), 131.
- Welch, G. F. J. C. V. A. R. G. (2020). Kalman filter. 1-3.
- Wen-shu, N., Liao-ni, W., & Qi, L. (2010). AHRS base on MEMS-IMU for aircraft model in wire-driven parallel suspension system. Paper presented at the 2010 International Conference on Mechanic Automation and Control Engineering.
- Zhi, Y., Li, G., Song, Q., Yu, K., & Zhang, J. (2017). Flight control law of unmanned aerial vehicles based on robust servo linear quadratic regulator and Kalman filtering. 14(1), 1729881416686952.

Cite this article: Unler, T., Guler, Y.S. (2025). Evaluating the Performance of Euler and Quaternion-Based AHRS Models in Embedded Systems for Aviation and Autonomous Vehicle Applications. *Journal of Aviation*, 9(2), 249-259.



This is an open access article distributed under the terms of the Creative Commons Attribution 4.0 International Licence

Copyright © 2025 Journal of Aviation <https://javsci.com> - <http://dergipark.gov.tr/jav>