

# Novel Non-Iterative Orientation Estimation for Wearable Motion Sensor Units Acquiring Accelerometer, Gyroscope, and Magnetometer Measurements

Aras Yurtman and Billur Barshan

Department of Electrical and Electronics Engineering  
Bilkent University, Bilkent, TR-06800 Ankara, Turkey  
yurtman@ee.bilkent.edu.tr, billur@ee.bilkent.edu.tr  
tel: (90-312) 290-2161, fax: (90-312) 266-4192

**Abstract** — We propose a novel non-iterative orientation estimation method based on the physical and geometrical properties of the acceleration, angular rate, and magnetic field vectors to estimate the orientation of motion sensor units. The proposed algorithm aims that the vertical (up) axis of the Earth coordinate frame is as close as possible to the measured acceleration vector and that the North axis of the Earth makes an angle with the detected magnetic field vector as close as possible to the estimated value of the magnetic dip angle. We obtain the sensor unit orientation based on the rotational quaternion transformation between the Earth and the sensor unit frames. We evaluate the proposed method by incorporating it into an activity recognition scheme for daily and sports activities which requires accurately estimated sensor unit orientations to achieve invariance to the orientations at which the units are worn on the body. Using four different classifiers on a publicly available dataset, the proposed methodology achieves an average activity recognition accuracy higher than the state-of-the-art methods, as well as being computationally efficient enough to be executed in real time.

**Keywords:** orientation estimation; quaternion; wearable motion sensors; accelerometer; gyroscope; magnetometer; inertial sensors; magnetic, angular rate, and gravity sensors

## 1 Introduction

Consumer-grade accelerometers, gyroscopes, and magnetometers have become prevalent in wearables such as smart watches, fitness trackers, motion capture systems, as well as in compact autonomous vehicles [1, 2]. Data acquired from these sensors can be exploited for accurate orientation estimation and tracking with applications in biomechanics, sports science, activity and gesture recognition, detection and classification of falls, virtual/augmented reality, dead reckoning, and navigation [3].

Determining the directions of any two non-parallel reference vectors is sufficient for estimating the orientation of the motion sensor unit at a given time instant. This is the concept of *vector matching*, which requires

that measurements of constant reference vectors are acquired. The reference vectors usually employed for this purpose are the gravity vector detected by the accelerometer and the magnetic field of the Earth measured by the magnetometer (Figure 1). However, neither can be acquired in isolation because the former is summed with the acceleration vector caused by the motion of the sensor unit while the latter is superposed with magnetic distortion, if any. Magnetic distortion can be caused by nearby ferromagnetic materials and constitutes a problem especially in man-made indoor environments. Hence, the *static* orientation estimate, calculated based on these two reference vectors, is not always accurate.

On the other hand, the *dynamic* orientation estimate in 3D is obtained by integrating the angular rate vector of a tri-axial gyroscope. This is not very reliable either, mainly because typically, the angular rate vector  $\omega$ , detected by the gyroscope, has an offset or bias error that varies with time and the temperature of the unit. It is a slowly varying additive offset, typically less than  $5^\circ/s$  or less than 1% of the gyroscope's full range, which needs to be subtracted from the gyroscope output at each time instant. Even though this bias can be compensated for to a certain extent through the use of error models [4], the orientation information obtained by integrating the rate output drifts over time and becomes unreliable in the long term. Since both the static and dynamic estimates are not very accurate and reliable on their own, these two estimates are often fused to improve the accuracy of the orientation estimate.

While most of the existing orientation estimation methods (OEMs) obtain the dynamic estimate by integrating the angular rate vector as described above, there are different techniques for estimating the static orientation in 3D.

Existing orientation estimation techniques can be classified into deterministic (least-squares), stochastic (Bayesian), and frequency-based approaches [5, 6]. The least-squares approach was originally introduced in Wahba's problem [7] which is a constrained least-squares optimization problem for finding the rotation matrix based on vector measurements acquired at a single time instant (single-frame method). Some variants of the deterministic single-frame estimation algorithms are: TRIAD (TRi-axial Attitude Determination) [8], FQA (Factored Quaternion Algorithm) [9], QUEST (QUaternion ESTimator) [8], GD (Gradient-Descent) [10], GN (Gauss-Newton) optimization [11], and LM (Levenberg-Marquardt) optimization [12] (LM is a blend of GD and GN.). All of these approaches are based on the concept of vector matching and can be used to solve Wahba's problem without the need for a prior estimate. TRIAD and FQA are simple and non-iterative algorithms that rely on geometric approaches to respectively calculate the rotation matrices and quaternions that represent the 3D orientation. Since they do not utilize gyroscopic angular rate measurements, a dynamic orientation estimate is not involved. These algorithms are mainly intended for static or slowly moving sensor units. On the other hand, GD, GN, and LM are iterative algorithms that first estimate the orientation by minimizing a cost function that decreases as the up and North directions of the Earth approach the acceleration and magnetic field vectors, respectively. The method proposed in [10] uses the GD algorithm to align the upward direction of the Earth frame with the acceleration vector and the North direction with the horizontal component of the magnetic field vector. To improve the computational efficiency, it uses an approximate solution and iterates the GD algorithm only once at each time sample. The algorithm proposed in [11] uses the GN method to make an alignment similar to that in [10]; however, unlike [10], it calculates the solution without making any approximations. The same study also provides a brief comparison between GD and GN algorithms for which the number of iterations is limited to ten and three, respectively. Based on the results, it is stated that GN is faster and does not require as many iterations as GD to reach the minimum point

and the estimated orientation angles do not fluctuate as much around their true values. These iterative algorithms may not always converge to the global minimum and are computationally intensive because they need to be iterated several times at each time sample. Once the static orientation estimate is obtained through a number of iterations, the static and dynamic orientation estimates are combined through weighted averaging at each time sample.

The most commonly used stochastic approach for dynamic orientation estimation is the Bayesian approach. Linear and extended Kalman filters, unscented Kalman filters, and more advanced Bayesian filters such as particle filters fall under this category. Kalman filters (KFs) represent a special class of algorithms for recursive Bayesian state estimation [13]. This type of filter aims to yield minimum mean-square error sequential estimates of the state vector using information about the motion dynamics. The state vector of the KF may be composed of 3D orientation and possibly other parameters and quantities of interest such as sensor bias levels. Another alternative is to construct a state vector of error processes (the indirect KF) [14]. Particle filters (PFs) are suitable for non-linear models where the noise model is not necessarily Gaussian distributed, unlike KFs [15]. A PF makes no assumptions regarding the linearity or the noise distribution of the system. Thus, PFs are applicable to a wide range of systems with any error model. However, the main disadvantage of the KF and PF approaches is their relatively high computational cost [16, 17, 18].

Existing OEMs, summarized above and reviewed in [5, 6] in more detail, are based on the assumption that the acceleration vector  $\mathbf{a}$  is along the vertical direction and the component of the magnetic field vector  $\mathbf{m}$  perpendicular to  $\mathbf{a}$  points to the North. However, the magnetic field of the Earth makes an angle with the horizontal plane called the *magnetic dip angle* which is approximately constant at a given location on the Earth in the short term. Since the existing OEMs do not make use of the value of the magnetic dip angle, they do not completely exploit the properties of the magnetic field vector. This may result in inaccurate orientation estimates when there is substantial acceleration besides gravity as well as considerable magnetic distortion in the vicinity. We propose to improve the static orientation estimate by using a non-iterative OEM based on the physical and geometrical properties of the detected acceleration and magnetic field vectors, both of which are completely involved in determining the vertical (up) and the North directions of the Earth. Our algorithm is designed such that it aims to simultaneously satisfy that the vertical axis of the Earth coordinate frame is as close as possible to the measured acceleration vector and that the North axis of the Earth makes an angle with the detected magnetic field vector as close as possible to the estimated value of the magnetic dip angle. We obtain the sensor unit orientation based on the rotational quaternion transformation between the Earth and the sensor unit frames. Thus, the main contribution of this article is to provide a novel non-iterative OEM which aims to improve the accuracy of the static orientation estimate. Since the true orientation information is not available for direct comparison, we evaluate our proposed method by integrating it into an existing daily and sports activity recognition scheme [19] that requires accurate sensor unit orientation estimates to achieve invariance to the orientation of the units through an orientation-invariant transformation (OIT).

The rest of this article is organized as follows: In the next section, we briefly introduce the notation used throughout the article and describe the representation of sensor unit orientation in 3D using quaternions. In Section 3, we explain the proposed methodology to improve the estimation of sensor unit orientation. We provide the details of the implementation of the existing state-of-the-art OEMs in Section 4. In Section 5, we provide information on the motion sensor unit and give a brief description of the instrumentation used in this study. Our

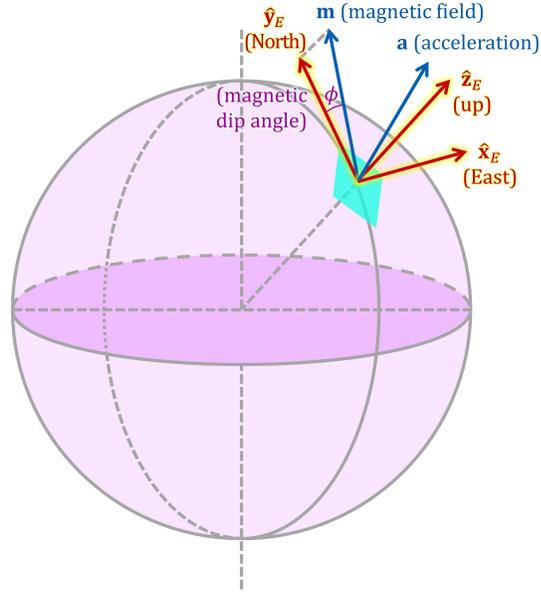


Figure 1: The Earth frame illustrated on an Earth model illustrating the unit vectors of the frame, the two reference vectors  $\mathbf{a}$  and  $\mathbf{m}$ , and the magnetic dip angle  $\phi$ .

earlier related work is summarized in Section 6. Section 7 provides a comparative evaluation between six existing OEMs and the proposed method within the context of activity recognition where accurate orientation estimation is highly desired. In the last section, we draw conclusions and provide some potential application areas for the proposed orientation estimation technique. It should be declared that this article reuses some content from the Ph.D. thesis of the first author [20] with permission.

## 2 Notation and Representation of Sensor Unit Orientation

Measurements acquired from the accelerometer, gyroscope, and magnetometer at time sample  $n$  are represented by  $3 \times 1$  vectors  $\mathbf{a}[n]$ ,  $\boldsymbol{\omega}[n]$ , and  $\mathbf{m}[n]$ , respectively. For a time segment of recorded measurements that contains  $N$  time samples, the discrete-time index  $n$  takes values between 1 and  $N$ , and is omitted for simplicity where needed. The hat notation is used for vectors normalized by their magnitudes (unit vectors):  $\hat{\mathbf{a}} \triangleq \mathbf{a}/\|\mathbf{a}\|$  and  $\hat{\mathbf{m}} \triangleq \mathbf{m}/\|\mathbf{m}\|$ , where  $\|\cdot\|$  denotes the Euclidean norm.

According to the East-North-Up (ENU) convention [21], the  $x, y, z$  axes of the Earth's coordinate frame  $E$  point in the East, North, and upward directions, respectively (Figure 1). The transformation between  $E$  and the sensor unit frame  $S$  at each time sample  $n$  can be represented by a  $3 \times 3$  rotation matrix  $\mathbf{R}[n]$  or equivalently by a  $4 \times 1$  rotation quaternion  $\mathbf{q}[n] = (q_1, q_2, q_3, q_4)^T$  [22] which satisfies the simple normalization constraint  $q_1^2 + q_2^2 + q_3^2 + q_4^2 = 1$ . The elements of the rotation matrix  $\mathbf{R}[n]$  can be expressed in terms of the four elements of the quaternion as follows [23]:

$$\mathbf{R}[n] = \begin{bmatrix} q_1^2 + q_2^2 - q_3^2 - q_4^2 & 2(q_2q_3 - q_1q_4) & 2(q_2q_4 + q_1q_3) \\ 2(q_2q_3 + q_1q_4) & q_1^2 - q_2^2 + q_3^2 - q_4^2 & 2(q_3q_4 - q_1q_2) \\ 2(q_2q_4 - q_1q_3) & 2(q_3q_4 + q_1q_2) & q_1^2 - q_2^2 - q_3^2 + q_4^2 \end{bmatrix} \quad (1)$$

The columns of  $\mathbf{R}[n]$  correspond to the unit vectors  $\hat{\mathbf{x}}_E, \hat{\mathbf{y}}_E, \hat{\mathbf{z}}_E$  of the Earth frame  $E$  with respect to the sensor unit frame  $S$ . Note that the transpose of  $\mathbf{R}[n]$  (which is the same as its inverse since  $\mathbf{R}[n]$  is an orthonormal matrix [24]), represents the orientation of the frame  $S$  with respect to the frame  $E$ .

Compared to rotation matrices, quaternions are more compact, more numerically stable, and more efficient. For these reasons, we have chosen to represent 3D orientation using quaternions in this study.

### 3 Proposed Methodology to Estimate the Sensor Unit Orientation

Given the current angular rate vector,  $\boldsymbol{\omega}[n] = (\omega_x[n], \omega_y[n], \omega_z[n])^T$ , the dynamic orientation  $\mathbf{q}_d[n]$  for  $n = 1, \dots, N$  is estimated based on the combined orientation estimate  $\mathbf{q}[n-1]$  at the previous time sample [see Equation (7)] and the augmented angular rate vector  $\boldsymbol{\omega}'[n] \triangleq (0, \omega_x[n], \omega_y[n], \omega_z[n])^T$  as

$$\mathbf{q}_d[n] = \mathbf{q}[n-1] + \Delta t \left( \frac{1}{2} \mathbf{q}[n-1] \otimes \boldsymbol{\omega}'[n] \right) \quad (2)$$

where the symbol  $\otimes$  denotes the quaternion product operator and  $\Delta t$  is the time step.

Assuming that the acceleration components resulting from the motion of the sensor unit average out to zero, gravity stands as the dominant component of  $\mathbf{a}$  in the long term. Consequently, averaging or low-pass filtering the acquired acceleration vectors provides an estimate of the direction of the gravity vector which points in the vertical direction of the Earth. Based on this assumption that the average of the  $\mathbf{a}$  vectors points to the vertical, we can estimate the magnetic dip angle  $\phi$  by averaging the angle between  $\mathbf{m}[n]$  and the horizontal plane (perpendicular to  $\mathbf{a}[n]$ ) over a short time segment:

$$\tilde{\phi} = \frac{1}{N} \sum_{n=1}^N \phi[n] \quad \text{where} \quad \phi[n] = \frac{\pi}{2} - \angle(\mathbf{a}[n], \mathbf{m}[n]) \quad (3)$$

Here,  $\angle(\cdot, \cdot) \in [0, \pi)$  denotes the angle between two 3D vectors and  $N$  is the number of time samples in a short time segment.

Our main purpose is to determine the vertical (up) direction and the magnetic North direction of the Earth based on the acquired accelerometer and magnetometer measurements so that we can estimate the 3D orientation of the sensor unit. The  $\hat{\mathbf{y}}_E$  and  $\hat{\mathbf{z}}_E$  unit vectors of the Earth frame need to be chosen in accordance with the directions of the two reference vectors  $\mathbf{a}$  and  $\mathbf{m}$ . If  $\tilde{\phi}$  were zero, then we could have taken the upward ( $\hat{\mathbf{z}}_E$ ) and the North ( $\hat{\mathbf{y}}_E$ ) axes in the same direction as the detected  $\mathbf{a}$  and  $\mathbf{m}$  vectors, respectively, as in some existing work. However, the  $\mathbf{m}$  vector makes an angle  $\phi$  with the horizontal which is different than zero in general so that it is not perpendicular to  $\mathbf{a}$ . In the one extreme case, if  $\hat{\mathbf{z}}_E$  is aligned with  $\mathbf{a}$  exactly, since  $\hat{\mathbf{y}}_E$  needs to be selected perpendicular to  $\hat{\mathbf{z}}_E$ , this means that the angle between  $\hat{\mathbf{y}}_E$  and  $\mathbf{m}$  will largely differ from  $\tilde{\phi}$  since the  $\phi[n]$  values are noisy. At the other extreme, if the angle between  $\hat{\mathbf{y}}_E$  and  $\mathbf{m}$  is set exactly equal to the estimated  $\tilde{\phi}$ , then the alignment between  $\hat{\mathbf{z}}_E$  and  $\mathbf{a}$  will not be achieved. In other words, if one condition is exactly satisfied, the other one exhibits a large error because the magnetic dip angle is noisy and different than zero in general. Thus, we propose three objectives that need to be satisfied:

- O1: align  $\hat{\mathbf{z}}_E$  with  $\hat{\mathbf{a}}$

O2: set the angle between  $\hat{\mathbf{y}}_E$  and  $\hat{\mathbf{m}}$  to the estimated magnetic dip angle  $\tilde{\phi}$

O3: select  $\hat{\mathbf{y}}_E$  and  $\hat{\mathbf{z}}_E$  perpendicular to each other

To satisfy O1 and O3 only (without O2), the up and North directions ( $\hat{\mathbf{z}}_E$  and  $\hat{\mathbf{y}}_E$ ) can be selected in the same direction as the  $\hat{\mathbf{a}}$  vector and the normalized component  $\hat{\mathbf{m}}_{\perp}$  of  $\mathbf{m}$  perpendicular to  $\hat{\mathbf{a}}$ , respectively, as in the TRIAD algorithm [8]:

$$\hat{\mathbf{m}}_{\perp} = \frac{\mathbf{m}_{\perp}}{\|\mathbf{m}_{\perp}\|} \quad \text{where} \quad \mathbf{m}_{\perp} = \mathbf{m} - (\hat{\mathbf{a}} \cdot \mathbf{m}) \hat{\mathbf{a}} \quad (4)$$

To satisfy O2 and O3 only (without O1), we may rotate the  $\hat{\mathbf{z}}_E$  and  $\hat{\mathbf{y}}_E$  axes on the  $\mathbf{a}$ - $\mathbf{m}$  plane about the axis  $\hat{\mathbf{m}} \times \hat{\mathbf{a}}$  by the angle

$$\alpha = \text{sign}(\mathbf{a} \cdot \mathbf{m}) (\angle(\mathbf{m}, \mathbf{m}_{\perp}) - |\tilde{\phi}|) = \text{sign}(\mathbf{a} \cdot \mathbf{m}) (\phi[n] - |\tilde{\phi}|) \quad (5)$$

where  $\text{sign}(\cdot)$  denotes the signum function. This rotation is depicted in Figure 2 for the two cases where the vectors  $\mathbf{a}$  and  $\mathbf{m}$  make acute and obtuse angles with each other.

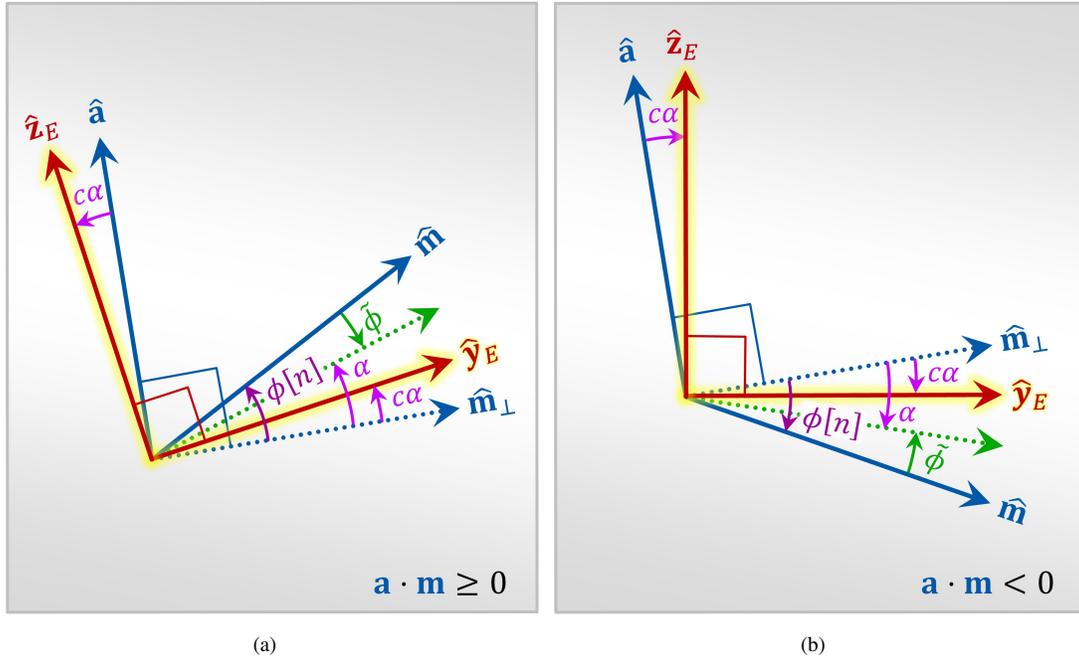


Figure 2: Selection of the  $\hat{\mathbf{z}}_E$  and  $\hat{\mathbf{y}}_E$  axes to estimate the static orientation for the cases where (a)  $\mathbf{a} \cdot \mathbf{m} \geq 0$  and (b)  $\mathbf{a} \cdot \mathbf{m} < 0$ .

Since the objectives O1 and O2 cannot be satisfied at the same time (unless  $\tilde{\phi} = 0$ ), in our solution to the problem, we geometrically determine  $\hat{\mathbf{y}}_E$  and  $\hat{\mathbf{z}}_E$  to satisfy each of the first two objectives as much as possible by making a compromise between them, while meeting the third objective exactly. We obtain the  $\hat{\mathbf{z}}_E$  and  $\hat{\mathbf{y}}_E$  directions by rotating the vectors  $\hat{\mathbf{a}}$  and  $\hat{\mathbf{m}}_{\perp}$  through an angle  $c\alpha$ , where  $c \in [0, 1]$  is a parameter of the algorithm:

$$\begin{aligned} \hat{\mathbf{z}}_E &= \hat{\mathbf{a}} \cos(c\alpha) - \hat{\mathbf{m}}_{\perp} \sin(c\alpha) \\ \hat{\mathbf{y}}_E &= \hat{\mathbf{a}} \sin(c\alpha) + \hat{\mathbf{m}}_{\perp} \cos(c\alpha) \end{aligned} \quad (6)$$

This way, we parametrize the amount of rotation and select the parameter  $c$  by parameter optimization as

described at the end of this section.

We select the remaining axis that points to the East as  $\hat{\mathbf{x}}_E = \hat{\mathbf{y}}_E \times \hat{\mathbf{z}}_E$  and represent the static orientation estimate by the quaternion  $\mathbf{q}_s[n]$  corresponding to the rotational transformation  $\mathbf{R}_s[n] = [\hat{\mathbf{x}}_E \ \hat{\mathbf{y}}_E \ \hat{\mathbf{z}}_E]$ .

We finally merge the dynamic and static estimates through weighted averaging to obtain the combined orientation estimate:

$$\mathbf{q}[n] = \mathcal{K} \mathbf{q}_d[n] + (1 - \mathcal{K}) \mathbf{q}_s[n] \quad (7)$$

where  $\mathcal{K} \in [0, 1]$  is the weight parameter of the algorithm. The flowchart of the algorithm is shown in Figure 3. Note that the proposed algorithm is not iterative unlike the OEMs GD, GN, and LM.

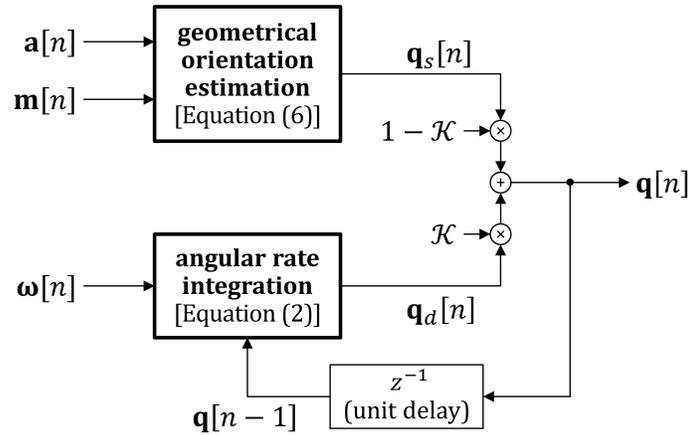


Figure 3: The flowchart of the proposed algorithm.

We optimize the parameters  $c$  and  $\mathcal{K}$  of the newly proposed OEM through a 2D grid search to maximize classification accuracy. On a coarse grid where both parameters vary between zero and one with 0.1 increments, the optimal values are  $(c^*, \mathcal{K}^*) = (0.40, 0.98)$ . On a fine grid where  $c \in \{0.30, 0.32, \dots, 0.80\}$  and  $\mathcal{K} \in \{0.90, 0.91, \dots, 1.00\}$ , the optimal parameter pair is  $(c^{**}, \mathcal{K}^{**}) = (0.36, 0.98)$ , which is the parameter pair used in this study. When  $c$  and  $\mathcal{K}$  are both set equal to zero, the proposed OEM reduces to the TRIAD algorithm.

## 4 Implementation of Existing OEMs and Their Initialization

We have considered six existing state-of-the-art orientation estimation algorithms (TRIAD, GD, GN, LM, KF, and PF) in this study for comparison with the proposed OEM in terms of accuracy and run time. The comparison is made under the same conditions as detailed below.

### 4.1 Iterative OEMs

In the GD-based OEM, we use a single, approximated GD iteration at each time sample, as in its original implementation [10]. We implement the GN and LM algorithms without imposing any limit to the number of iterations, and terminate them when the change in the cost function is smaller than  $10^{-3}$ . For LM, we use the algorithm provided on page 438 in [12]: We initialize the damping parameter  $\lambda$  with 0.5 for the first iteration and adaptively change it by using the multiplier  $\nu = 2$  in the iterations that follow.

We initialize the iterative OEMs and the proposed OEM as follows: Because of the dependence of the dynamic orientation estimate  $\mathbf{q}_d[n]$  on the combined estimate  $\mathbf{q}[n-1]$  at the previous time sample and since such a combined estimate is not available at the first time sample, Equation (2) is not evaluated for  $n = 1$ . Thus, the combined estimate in Equation (7) at  $n = 1$  is calculated solely based on the static estimate without using the dynamic estimate:  $\mathbf{q}[1] = \mathbf{q}_s[1]$ . The iterative methods (GD, GN, and LM) are executed at each time sample  $n$  to estimate the static orientation  $\mathbf{q}_s[n]$ . Since there is no information about the orientation at  $n = 1$ , they are initialized with the quaternion estimated by the TRIAD algorithm at the first time sample. (Note that both TRIAD and the proposed algorithm can already make an orientation estimate at the very first time sample.) For  $n = 2, \dots, N$ , the combined orientation estimate  $\mathbf{q}[n-1]$  at the previous time sample is used as the initial condition [10, 11]. We apply the OEMs to each time segment (of 5 s duration) of the recorded data separately.

## 4.2 Indirect Complementary Kalman Filter

As a commonly employed stochastic approach, we have implemented an indirect complementary Kalman Filter (KF) where the term indirect means that the KF operates on the error vector rather than the state vector itself [25]. The term complementary indicates that the KF balances orientation estimates coming from (i) the accelerometer and the magnetometer and (ii) from the gyroscope [25]. The KF uses a conventional iterative prediction/correction cycle and adaptively combines the static and dynamic orientation estimates. (All the remaining OEMs, except for TRIAD, also combine the static and dynamic orientation estimates.) The error process is modeled through the  $12 \times 1$  state vector:

$$\mathbf{x}_n = \begin{bmatrix} \boldsymbol{\theta}_{\varepsilon,n} \\ \mathbf{b}_{\varepsilon,n} \\ \mathbf{a}_{\varepsilon,n} \\ \mathbf{m}_{\varepsilon,n} \end{bmatrix} \quad (8)$$

where  $\boldsymbol{\theta}_{\varepsilon,n}$  is the  $3 \times 1$  orientation error vector,  $\mathbf{b}_{\varepsilon,n}$  is the  $3 \times 1$  gyroscope zero-rate offset vector (or bias vector),  $\mathbf{a}_{\varepsilon,n}$  is the  $3 \times 1$  acceleration error vector with respect to the sensor unit frame, and  $\mathbf{m}_{\varepsilon,n}$  is the  $3 \times 1$  magnetic disturbance error vector with respect to the sensor unit frame, all calculated at the time sample  $n$  [25]. The gyroscope zero rate offset vector  $\mathbf{b}_{\varepsilon,n}$  models the output of the tri-axial gyroscope when the sensor unit is not rotating but stationary.

The  $6 \times 1$  measurement vector is defined as:

$$\mathbf{z}_n = \begin{bmatrix} \mathbf{g}_{d,n} - \mathbf{g}_{s,n} \\ \mathbf{m}_{d,n} - \mathbf{m}_{s,n} \end{bmatrix} \quad (9)$$

where  $\mathbf{g}_{d,n}$  and  $\mathbf{g}_{s,n}$  are the dynamic and static estimates of the gravity vector at time sample  $n$ , whereas  $\mathbf{m}_{d,n}$  and  $\mathbf{m}_{s,n}$  are their counterparts for the magnetic field of the Earth [25]. Both the process and the measurement models of the KF are linear and expressed as:

$$\mathbf{x}_n = \mathbf{A}_n \mathbf{x}_{n-1} + \mathbf{w}_n \quad (10)$$

$$\mathbf{z}_n = \mathbf{H}_n \mathbf{x}_n + \mathbf{v}_n \quad (11)$$

Here,  $\mathbf{w}_n$  and  $\mathbf{v}_n$  are additive, zero-mean, mutually-independent Gaussian noise vectors,  $\mathbf{A}_n = \mathbf{0}$ , and  $\mathbf{H}_n$  is a matrix calculated based on the dynamic orientation estimate [25]. This linear KF, inherently recursive, is iterated once at each time sample  $n$  when a new set of measurements is acquired. We have implemented the KF using the function `ahrsfilter` that is available in the Sensor Fusion and Tracking Toolbox of MATLAB R2018b [26] and tuned its input noise parameters using the information provided by the manufacturer of the motion sensor unit in Table A.1 of the Appendix. This is done by multiplying the square of the noise density with the bandwidth provided in the table. After changing the units deg/s to rad/s (for the gyroscope) and mGauss to  $\mu\text{T}$  (for the magnetometer), we obtain the noise variances of the accelerometer, gyroscope, and magnetometer as  $0.00012 (\text{m/s}^2)^2$ ,  $3.046 \times 10^{-5} (\text{rad/s})^2$ , and  $0.025 (\mu\text{T})^2$  [27], respectively. To tune the input parameters of the KF, we have optimized the noise variances through a 3D grid search where we have multiplied each by the factors 0.1, 0.5, 1.0, 1.5, 2.0, 2.5, 3.0, 3.5, and 4.0, and considered all  $9^3 = 729$  combinations of these parameters. Including all the earlier trials that we have made, we can say that we have tested more than 1000 combinations of noise parameters. The highest classification accuracy of 79.9% is obtained when the noise variance values given above are multiplied by 1.5, 0.5, and 2.5, respectively. We took the expected magnetic field strength parameter of the KF as  $47.2583 \mu\text{T}$ , which is the value around the time when the experimental data were acquired at Bilkent University according to the International Geomagnetic Reference Field model [28]. Input sampling rate of sensor data is taken as 25 Hz. The orientation format at the output of the KF is chosen to be in quaternion form. Default values are used for the remaining seven parameters of the KF. [26].

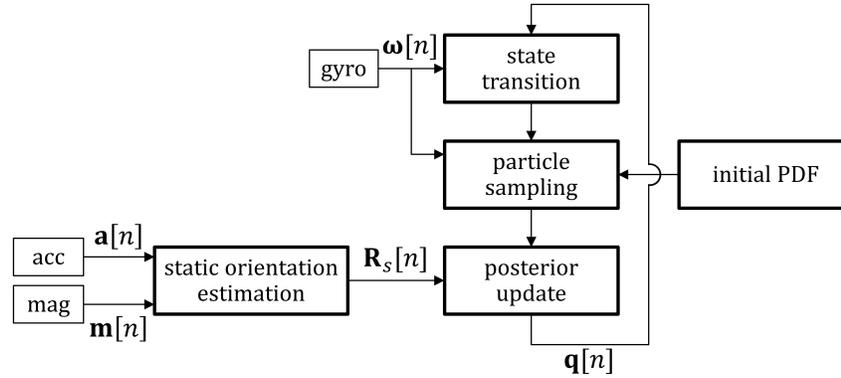


Figure 4: Overview of the particle filter approach used for 3D orientation estimation.

### 4.3 Particle Filter Based Orientation Estimation

Particle filters track the posterior of a system by using a number of particles to approximate the underlying probability distribution [15]. This eliminates the need to make assumptions regarding the noise distributions and the motion model. In our application of 3D orientation estimation, the state vector of the PF that we have implemented represents the combined orientation estimate  $\mathbf{q}[n]$  that is dynamically updated at each time sample  $n$  [29]. An overview of the PF approach is provided in Figure 4.

The static estimate is adaptively merged with the dynamic one through the following three steps: (1) The

combined orientation estimate  $\mathbf{q}[n-1]$  at the previous time sample is integrated using the measured angular rate vector  $\boldsymbol{\omega}[n]$  at the current time sample (state transition). (2) Zero-mean Gaussian noise is added to the angular rate vector  $\boldsymbol{\omega}[n]$  using the pre-known mean and variance parameters to model the effects of gyroscope process noise.  $M$  quaternion particles are sampled from the state space at each time sample (particle sampling). Each quaternion particle  $\mathbf{q}_p^{(j)}[n]$  is assigned a prior weight  $w_{\text{prior}}^{(j)}[n]$  using a weighting function  $\mathcal{W}$  such that  $w_{\text{prior}}^{(j)}[n] = \mathcal{W}(\mathbf{q}_p^{(j)}[n])$ , where  $j = 1, \dots, M$  is the particle index. (3) The acquired measurements from the accelerometer and the magnetometer are used to estimate the static orientation using a method that is equivalent to the TRIAD algorithm (static orientation estimation) [29] and the rotation matrix  $\mathbf{R}_s[n]$  corresponding to the static orientation estimate is calculated using Equation (1). The difference matrix between  $\mathbf{R}_s[n]$  and the rotation matrix  $\mathbf{R}_p^{(j)}[n]$  corresponding to each quaternion particle is calculated:

$$\mathbf{R}_{\text{diff}}^{(j)}[n] \triangleq \mathbf{R}_s[n] - \mathbf{R}_p^{(j)}[n] = \begin{bmatrix} \mathbf{x}_{\text{diff}}^{(j)}[n] & \mathbf{y}_{\text{diff}}^{(j)}[n] & \mathbf{z}_{\text{diff}}^{(j)}[n] \end{bmatrix} \quad (12)$$

Based on the difference matrix in Equation (12), the likelihood is calculated for each quaternion particle  $\mathbf{q}_p^{(j)}[n]$ :

$$L^{(j)}[n] \propto \frac{1}{\|\mathbf{x}_{\text{diff}}^{(j)}[n]\| \cdot \|\mathbf{y}_{\text{diff}}^{(j)}[n]\| \cdot \|\mathbf{z}_{\text{diff}}^{(j)}[n]\|} \quad (13)$$

Finally, the posterior probabilities are updated based on the prior probabilities and the likelihood of each quaternion particle as (posterior update):

$$w_{\text{post}}^{(j)}[n] = \frac{w_{\text{prior}}^{(j)}[n] \cdot L^{(j)}[n]}{\mathcal{P}} \quad (14)$$

where  $\mathcal{P}$  is the conditional probability for scaling. The combined quaternion estimate is calculated through weighted averaging:

$$\mathbf{q}[n] = \sum_{j=1}^M w_{\text{post}}^{(j)}[n] \cdot \mathbf{q}_p^{(j)}[n] \quad (15)$$

In our implementation of the PF, we have employed  $M = 1000$  quaternion particles.

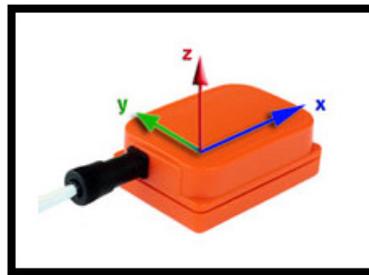


Figure 5: The Xsens MTx sensor unit [27].

## 5 Description of the Motion Sensor Unit and Instrumentation

We have used a consumer-grade inertial/magnetic measurement unit (IMMU), manufactured under the model name MTx by Xsens Technologies in the Netherlands [27]. The MTx unit contains three tri-axial sensors: an accelerometer, a gyroscope, and a magnetometer that acquire 3D acceleration, rate of turn (angular velocity vector), and the Earth's magnetic field vector, respectively. Each motion sensor unit is programmed via an interface program called MT Manager to capture the raw or calibrated data with a sampling frequency of up to 512 Hz. General specifications of the unit provided by the manufacturer are provided in Table A.1 in the Appendix.

We have used a sensor configuration with five MTx motion sensor units in this study (see Section 6). Accelerometers of two of the units can sense up to  $\pm 5g$  and the other three can sense within the range  $\pm 18g$ , where  $g = 9.80665 \text{ m/s}^2$  is the standard gravitational constant. The gyroscope in the MTx unit can detect angular velocities in the range of  $\pm 1200^\circ/\text{sec}$ ; the magnetometer can sense magnetic fields within the range  $\pm 75 \mu\text{T}$ . We use all three types of sensory data in all three dimensions.

The five MTx units are connected with 1 m cables to a device called the Xbus Master that can synchronize up to ten MTx units. The Xbus Master is attached to the subject's belt and transmits data from the five MTx units to the receiver using a Bluetooth<sup>TM</sup> connection. The receiver is connected to a laptop computer via a USB port. Two of the five MTx units are directly connected to the Xbus Master while the remaining three are indirectly connected to the Xbus Master by wires to the other two.

## 6 Related Earlier Work

In [19, 20], we proposed a methodology for recognizing daily and sports activities that requires accurate sensor unit orientation estimates to allow the units to be worn on the body at any orientation. This methodology relies on accurate estimation of the orientation of the sensor units with respect to the Earth frame based on the data they acquire. In that study, we employed the GN algorithm [11] to estimate the orientation of the sensor units. Here, we demonstrate that the activity recognition accuracy can be considerably improved by only replacing the GN algorithm with the newly proposed OEM. Hence, our main motivation is to achieve improvement in the accuracy of 3D orientation estimation and exploit this improvement in applications that require accurate orientation estimation in 3D. One potential application in the field of biomechanics would be estimating the 3D orientation of the human body parts [5].

We use a publicly available dataset acquired by our research group, comprised of 19 daily and sports activities which is also accessible through IEEE Data Port [30, 31, 32]. The dataset contains the following activities: sitting, standing, lying on back and on right side, ascending and descending stairs, standing still in an elevator, moving around in an elevator, walking in a parking lot, walking on a treadmill in flat and  $15^\circ$  inclined positions at a speed of 4 km/h, running on a treadmill at a speed of 8 km/h, exercising on a stepper, exercising on a cross trainer, cycling on an exercise bike in horizontal and vertical positions, rowing, jumping, and playing basketball.

Activities are performed by eight anonymous subjects who wear the five motion sensor units (on the chest, both wrists, and both knees). Each subject performs each activity for five minutes. We acquire data from each axis of each sensor unit using a sampling frequency of 25 Hz which is sufficient to capture human motion during daily

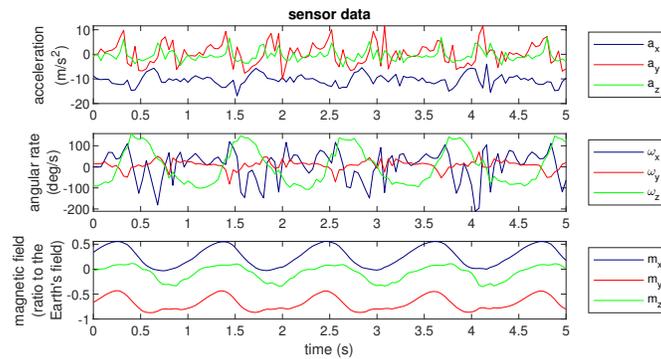


Figure 6: Raw sensor data acquired from the nine axes of the sensor unit over one 5 s time segment.

and sports activities. Dividing each 5 min recording into 5 s non-overlapping time segments results in a total of 9,120 (= 60 time segments per 5 min recording  $\times$  19 activities  $\times$  8 subjects) such time segments. Figure 6 shows the raw data acquired from the sensor unit on the right leg of a subject over one such 5 s time segment during the activity of walking on a flat treadmill.

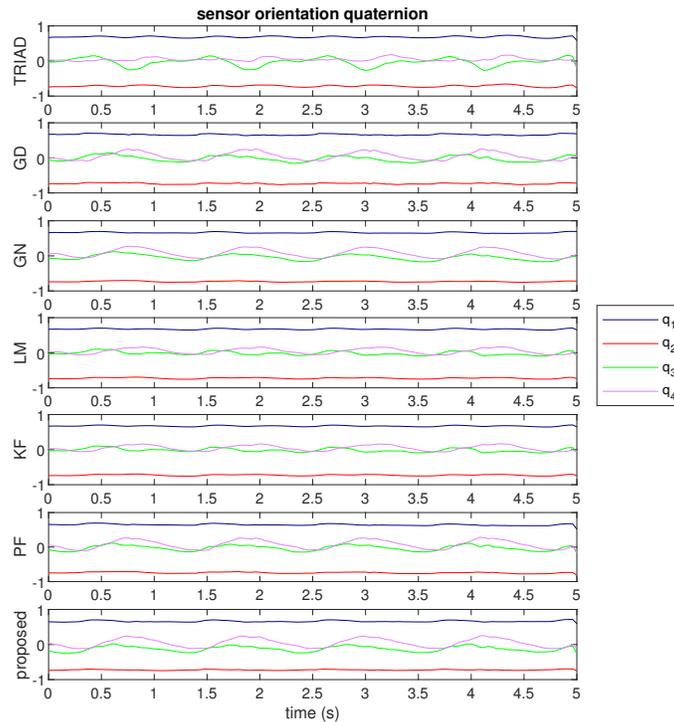


Figure 7: The four estimated elements of the orientation quaternions plotted as a function of time over one 5 s time segment for each of the OEMs considered and compared in this study.

## 7 Comparative Evaluation of the Proposed and Existing OEMs

As stated in Section 4, we estimate the sensor unit orientation by using six existing OEMs (TRIAD, GD, GN, LM, KF, and PF) and the newly developed non-iterative method that we propose here. The four estimated elements of the quaternions  $\mathbf{q}[n]$  representing the sensor unit orientations using the existing and proposed algorithms are

plotted as a function of time over one 5-s time segment in Figure 7 based on the raw data illustrated in Figure 6. It can be observed from the figure that the estimates are consistent with each other. Since the true orientation information (the ground truth) is not available for direct comparison, we evaluate the proposed method indirectly by incorporating it into an activity recognition scheme for daily and sports activities which requires accurately estimated sensor unit orientations to achieve invariance to the orientations at which the units are worn on the body.

We have implemented nine approaches for activity recognition: The REF method is the standard activity recognition scheme with sensor units fixed to the body at proper orientations and does not transform the acquired data in any way [19, 32]. In ROT, we simulate arbitrarily oriented sensor units by randomly rotating the acquired data vectors through a rotational transformation, independently generated for each time segment of each sensor unit [19]. The OIT approach allows the units to be fixed to the body at any orientation by representing the acquired data in the Earth frame  $E$  together with the use of differential quaternions [19]. The OIT requires accurate estimation of the sensor unit orientation. The seven variants of the OIT using the existing and proposed OEMs are respectively denoted by OIT-TRIAD, OIT-GD, OIT-GN, OIT-LM, OIT-KF, OIT-PF, and OIT-proposed.

Next, we closely follow the activity recognition scheme in [19], which involves the basic stages of feature extraction, feature reduction, feature normalization, and classification of the (possibly transformed) data. We extract the following statistical features per each 5 s time segment of each axis of each sensor type of the (possibly transformed) data: minimum, maximum, mean, variance, skewness, kurtosis, ten coefficients of the autocorrelation sequence, and the five largest discrete Fourier transform (DFT) peaks with the corresponding frequencies, resulting in a total of 26 features per time segment of each axis of each sensor type. We normalize the features to the interval  $[0, 1]$  for each subject. For the REF approach that does not involve any kind of transformation, there exist 1,170 ( $= 5$  sensor units  $\times 9$  axes  $\times 26$  features per axis) features that are stacked to form a 1,170-element feature vector per time segment. For the OIT approach described in [19], 13 axes are used instead of nine, and thus, there exist 1,690 features instead of 1,170. We reduce the total number of features from 1,170 to 30 for REF and ROT and from 1,690 to 30 for all the variants of OIT through the use of principal component analysis.

In [19], we have considered seven classifiers among which Support Vector Machines (SVM) usually showed outstanding performance, followed by Linear Discriminant Classifier (LDC), Artificial Neural Networks (ANN), and Bayesian Decision Making (BDM). In this study, we limit the number of classifiers to these best-performing four, select their parameters as in [19], and evaluate their accuracies through leave-one-subject-out cross validation. In this type of cross validation, one subject's data are left out while training the classifier with the remaining subjects' data. The left out subject's data are then used for testing. This process is repeated for each subject.

Activity recognition accuracies for the nine approaches that use the four selected classifiers are provided in Figure 8. As expected, the highest accuracy is obtained with REF that uses properly oriented sensor units and the lowest with ROT where the units are randomly oriented without the use of any kind of OIT. All seven OEMs, when integrated into the OIT, improve the accuracy compared to ROT. However, the proposed OEM is superior to the other six, achieving an average accuracy 8.0%, 4.5%, 4.3%, 4.2%, 4.6%, and 6.0% higher than OIT-TRIAD, OIT-GD, OIT-GN, OIT-LM, OIT-KF, and OIT-PF, respectively (Figure 8(b)). Compared to REF, the average accuracy of OIT-proposed is 2.6% lower, which is naturally expected. The thin horizontal sticks in parts (a) and (b) of the figure indicate plus/minus one standard deviation over the cross-validation iterations of a particular classifier and

over the four classifiers, respectively.

Referring to Figure 8(a), SVM usually performs the best among the four classifiers, demonstrating its robustness to variations in the data. For five of the seven variants of the OIT, it achieves an accuracy noticeably higher than the remaining classifiers. LDC is the second best classifier on the average. The BDM classifier, when used with six of the seven variants of OIT (all except OIT-TRIAD), unexpectedly obtains an accuracy higher than REF, despite that the sensor units are allowed to be arbitrarily oriented on the body at pre-determined positions.

## 8 Run-Time Analysis

We have determined the run times of the OEMs by running them stand alone (that is, not as part of an OIT but externally and independently from the activity recognition scheme). According to the run times provided in Table 1, the proposed OEM is computationally more efficient than GN, LM, KF, and PF by factors of 2.9, 5.8, 4.6, and 200.7, and less efficient than TRIAD and GD by factors of 1.02 and 1.5, respectively. Since the computationally efficient approaches (GD and TRIAD) are not very accurate and the slightly more accurate algorithms (GN and LM) have much longer run times, the newly proposed method achieves a satisfactory compromise between accuracy and run time. For comparison, it is stated in [18] that linear and extended Kalman filter based approaches take 3.1 and 5.5 times more processing, respectively, compared to the approximated GD as in [10]. Our result for the run-time of the indirect complementary KF (which represents a linear model) can be considered to be consistent with this statement, since it is larger than the run time of the approximated GD by a factor of 6.8. The larger factor of 6.8 that we obtained (compared to the reported 5.5) could have been caused by the different models used for the state and measurement processes and other differences in implementing Kalman filters. The average classification times of the four classifiers are 0.39, 0.04, 0.01, and 1.53 ms per time segment, which can be neglected compared to the run times of the OEMs.

OEM	run time per 5 s time segment (ms)
TRIAD	19.45
GD	13.36
GN	57.66
LM	115.93
KF	91.13
PF	3977.67
proposed	19.82

Table 1: Average run times of the OEMs compared in this study.

## 9 Conclusions

We have demonstrated that among the six state-of-the-art OEMs, the simpler and computationally efficient TRIAD, GD, KF, and PF are not very accurate (within the context of activity recognition) whereas GN, LM, KF, and PF have high computational cost, GN and LM being slightly more accurate than TRIAD, GD, KF, and PF. We have developed a non-iterative OEM based on physical and geometric properties of two reference vectors that is simple

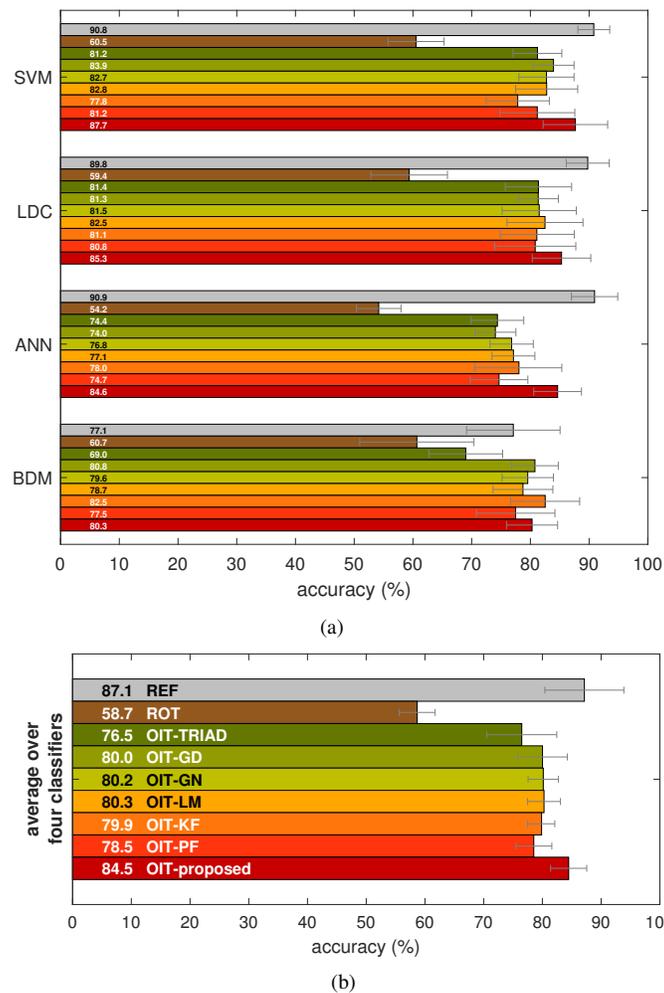


Figure 8: Activity recognition accuracy for the data transformation techniques and classifiers. (a) Individual results of the four selected classifiers and (b) their average accuracy.

to implement and efficient for real-time execution. We have evaluated the effectiveness of our method under dynamic conditions in a real-world scenario of daily and sports activity recognition where the motion sensor units can be worn on the body at arbitrary orientations. By only replacing the OEM in this scheme with the newly proposed one, the classification accuracy is improved and the run time is considerably reduced. Our proposed orientation estimation method can be considered to be universal in the sense that it can be employed for estimating the orientation of any motion sensor unit or IMMU of any grade (e.g., consumer, industrial, tactical, navigation, marine grade) that provides tri-axial data from an accelerometer, a gyroscope, and a magnetometer, corresponding to the three components of the acceleration, angular velocity, and magnetic field vectors. Since the acceleration vector and the components of the magnetic field vector are involved in the estimation process as reference vectors, the proposed methodology would work anywhere on or close to the surface of the Earth where the total acceleration vector (including the gravity component) and the magnetic field vector of the Earth can be detected by the motion sensor unit. If gyroscopic data are not available, as in gyro-free aiding systems, our method can still provide an accurate static orientation estimate through geometric orientation estimation at each time sample. More generally, according to the concept of vector matching, it is sufficient to have any two non-

parallel reference vectors detectable by the IMMU for estimating the orientation of the sensor unit. This means that the proposed methodology is also applicable at altitudes or depths (not necessarily so close to the Earth’s surface) where different reference values and, more importantly, reference directions for the gravity vector and the magnetic field vector are detectable, the latter possibly with different magnetic dip angle values. However, the method may not provide very accurate results if the acceleration components resulting from the motion of the sensor unit do not average out to zero or if there is considerable amount of magnetic distortion in the environment. In future work, the proposed OEM can be directly compared with an absolute reference that provides the ground truth for the 3D orientation estimate. The proposed OEM can be employed in other applications of wearables where the orientations of motion sensor units need to be estimated, such as in estimating the orientation of human body segments in biomechanics (e.g., for posture and gait analysis), sports science, rehabilitation (for orthotics, prosthetics, and exoskeletons), virtual/augmented reality, and fall detection.

## Appendix: Manufacturer’s Specifications for the MTx Motion Sensor Unit

The manufacturer’s specifications for the MTx unit are provided below. Additional details are available in [27].

Xsens MTx sensor unit	accelerometer	gyroscope	magnetometer	temperature
units	m/s <sup>2</sup>	deg/s	mGauss	°C
dimensions	3 axes	3 axes	3 axes	–
full scale	±50 and ±180	±1200	±750	–55 to +125
linearity (% of full scale)	0.2	0.1	0.2	< 1
bias stability	0.02	1	0.1	0.5
scale factor stability	0.03	–	0.5	–
noise density (units/√Hz)	0.002	0.05	0.5	–
alignment error (deg)	0.1	0.1	0.1	–
bandwidth (Hz)	30	40	10	–
A/D resolution (bits)	16	16	16	12
maximum data rate (Hz)	512			

Table A.1: General specifications of the Xsens MTx unit [27] provided by the manufacturer.

## References

- [1] S. Ghasemi-Moghadam and M. R. Homaeinezhad, “Attitude determination by combining arrays of MEMS accelerometers, gyros, and magnetometers via quaternion-based complementary filter,” *International Journal of Numerical Modelling*, vol.31, no.3, May/June 2018.
- [2] G. Seçer and B. Barshan, “Improvements in deterministic error modeling and calibration of inertial sensors and magnetometers,” *Sensors and Actuators A: Physical*, vol.247, pp.522–538, 15 August 2016.
- [3] T. Michel, P. Genevès, H. Fourati, and N. Layaïda, “On attitude estimation with smartphones,” *Proc. IEEE Int. Conf. on Pervasive Computing and Communications (PerCom)*, Kona, HI, U.S.A., 13–17 March 2017, pp.267–275.
- [4] B. Barshan and H. F. Durrant-Whyte, “Evaluation of a solid-state gyroscope for robotics applications,” *IEEE Transactions on Instrumentation and Measurement*, vol.44, no.1, pp.61–67, February 1995.

- [5] A. M. Sabatini, "Estimating three-dimensional orientation of human body parts by inertial/magnetic sensing," *Sensors*, vol.11, no.2, pp.1489–1525, 2011.
- [6] Y. Tian, H. Wei, and J. Tan, "An adaptive-gain complementary filter for real-time human motion tracking with MARG sensors in free-living environments," *IEEE Transactions on Neural Systems Rehabilitation Engineering*, vol.21, no.2, pp.254–264, March 2013.
- [7] G. Wahba, "A least-square estimate of spacecraft attitude," *SIAM Review*, vol.7, no.3, p.409, 1965.
- [8] M. D. Shuster and S. D. Oh, "Three-axis attitude determination from vector observations," *Journal of Guidance, Control, and Dynamics*, vol.4, no.1, pp.70–77, 1981.
- [9] X. Yun, E. R. Bachmann, and R. B. McGhee, "A simplified quaternion-based algorithm for orientation estimation from Earth gravity and magnetic field measurements," *IEEE Transactions on Instrumentation and Measurement*, vol.57, no.3, pp.638–650, March 2008.
- [10] S. O. H. Madgwick, A. J. L. Harrison, and R. Vaidyanathan, "Estimation of IMU and MARG orientation using a gradient descent algorithm," *Proc. IEEE Int. Conf. on Rehabilitation Robotics*, Zürich, Switzerland, 29 June–1 July 2011.
- [11] D. Comotti, "Orientation estimation based on Gauss-Newton method and implementation of a quaternion complementary filter," Technical Report, Dept. of Comp. Sci. and Eng., Univ. Bergamo, Italy, 2011.
- [12] D. W. Marquardt, "An algorithm for least-squares estimation of nonlinear parameters," *Journal of the Society for Industrial and Applied Mathematics*, vol.11, no.2, pp.431–441, June 1963.
- [13] R. C. Brown, *Introduction to Random Signal Analysis and Kalman Filtering*, John Wiley & Sons, New York, NY, U.S.A., 1983.
- [14] P. S. Maybeck, *Stochastic Models, Estimation and Control*, Academic Press, New York, NY, U.S.A., 1982.
- [15] A. Doucet and A. M. Johansen, "A tutorial on particle filtering and smoothing: Fifteen years later," In *Handbook of Nonlinear Filtering*, editors: D. Crisan and B. Rozovsky, Cambridge University Press, Cambridge, U.K., 2009.
- [16] M. Ghobadi, P. Singla, and E. T. Esfahani, "Robust attitude estimation from uncertain observations of inertial sensors using covariance inflated multiplicative extended Kalman filter," *IEEE Transactions on Instrumentation and Measurement*, vol.67, no.1, pp.209–217, January 2018.
- [17] H. Admed and M. Tahir, "Improving the accuracy of human body orientation estimation with wearable IMU sensors," *IEEE Transactions on Instrumentation and Measurement*, vol.66, no.3, pp.535–542, March 2017.
- [18] R. G. Valenti, I. Dryanovski, and J. Xiao, "A linear Kalman filter for MARG orientation estimation using the algebraic quaternion algorithm," *IEEE Transactions on Instrumentation and Measurement*, vol.65, no.2, pp.467–481, February 2016.

- [19] A. Yurtman, B. Barshan, and B. Fidan, "Activity recognition invariant to wearable sensor unit orientation using differential rotational transformations represented by quaternions," *Sensors*, vol.18, no.8, article no: 2725, August 2018.
- [20] A. Yurtman, *Activity Recognition Invariant to Position and Orientation of Wearable Motion Sensor Units*, Ph.D. thesis, Department of Electrical and Electronics Engineering, Bilkent University, Ankara, Turkey, April 2019.
- [21] G. Cai, B. M. Chen, and T. H. Lee, "Coordinate Systems and Transformations," *Unmanned Rotorcraft Systems*, Chapter 2, pp.23-34, Springer, London, U.K., 2011.
- [22] J. B. Kuipers, *Quaternions and Rotation Sequences*, Princeton University Press, NJ, U.S.A., 1999.
- [23] J. Diebel, "Representing attitude: Euler angles, unit quaternions, and rotation vectors," Technical Report, Dept. of Aeronautics and Astronautics, Stanford University, 2006. <http://www.swarthmore.edu/NatSci/mzucker1/papers/diebel2006attitude.pdf>
- [24] G. Strang, *Introduction to Linear Algebra*, 5th edition, Wellesley-Cambridge Press, Wellesley, MA, U.S.A., 2016.
- [25] M. Pedley, M. Stanley, and Z. Baranski, "Freescale Sensor Fusion Kalman Filter," Open Source Sensor Fusion, GitHub Repository, October 2014. Available from: <https://github.com/memsindustrygroup/Open-Source-Sensor-Fusion>
- [26] Sensor Fusion and Tracking Toolbox, MATLAB R2018b, The MathWorks, Inc., Natick, MA, U.S.A.
- [27] Xsens Technologies B.V., Enschede, The Netherlands, *MTi, MTx, and XM-B User Manual and Technical Documentation*, 2019, <http://www.xsens.com>
- [28] E. Thébault et al., "International Geomagnetic Reference Field: the 12th generation," *Earth, Planets, and Space*, vol.67, no.79, 27 May 2015.
- [29] N. Yadav and C. Bleakley, "Accurate orientation estimation using AHRS under conditions of magnetic distortion," *Sensors*, vol.14, no.11, pp.20008–20024, October 2014.
- [30] K. Altun and B. Barshan, "Daily and Sports Activities Dataset," IEEE Data Port, February 2019. (DOI: <http://dx.doi.org/10.21227/at1v-6f84>)
- [31] K. Altun and B. Barshan, "Daily and Sports Activities Dataset," UCI Machine Learning Repository, Univ. California, Irvine, School of Information and Computer Sciences, 2013. (<http://archive.ics.uci.edu/ml/datasets/Daily+and+Sports+Activities>)
- [32] K. Altun, B. Barshan, and O. Tunçel, "Comparative study on classifying human activities with miniature inertial and magnetic sensors," *Pattern Recognition*, vol.43, no.10, pp.3605–3620, October 2010.

## Vitae:



**Aras Yurtman** received B.S., M.S., and Ph.D. degrees in 2010, 2012, 2019 from Bilkent University, Ankara, Turkey, all in electrical and electronics engineering. He served as a research and teaching assistant and as an administrative assistant at the same department during his Ph.D. studies. His work has been published in journals such as *Computer Methods and Programs in Biomedicine*, *Applied Artificial Intelligence*, *The Computer Journal*, and *Sensors*. His research interests are in wearable sensing, human motion analysis, sensor signal processing, instrumentation and measurement, machine learning, and pattern recognition.



**Billur Barshan** received B.S. degrees in both electrical engineering and physics from Boğaziçi University in Istanbul, Turkey in 1986 and M.S. and Ph.D. degrees in electrical engineering from Yale University in the U.S.A. in 1988 and 1991, respectively. Dr. Barshan was a research assistant at Yale from 1987 to 1991 and a postdoctoral researcher at the Robotics Research Group at the University of Oxford in the U.K. from 1991 to 1993. In 1993, she joined the faculty of Bilkent University in Ankara, Turkey, where she is a professor in the Department of Electrical and Electronics Engineering. Dr. Barshan received the TÜBİTAK Incentive Award (1998), METU Mustafa Parlar Foundation Research Award (1999), and two best paper awards. She served on the Management Committee of the COST-IC0903 Action MOVE between 2010 and 2013. Dr. Barshan's current research interests include wearable sensing, intelligent sensing, instrumentation and measurement, detection and classification of falls, machine learning, and multi-sensor data fusion.