IEEE
Sensors Council

Estimating Relative Angles Using Two Inertial Measurement Units Without Magnetometers

Seung Yun Song, Yinan Pei, *Student Member, IEEE*, and Elizabeth T. Hsiao-Wecksler, *Member, IEEE*

Abstract—**Inertial measurement units (IMUs) are used in biomechanical and clinical applications for quantifying joint kinematics. This study aimed to assist researchers new to IMUs and wanting to develop an inexpensive IMU system to estimate the relative angle between IMUs, while understanding the different algorithms for estimating angular kinematics. Thus, there were three subgoals: 1) to present a low-cost and convenient IMU system utilizing two 6-axis IMUs for computing the relative angle between the IMUs; 2) to examine seven methods for estimating the angular kinematics of an IMU; and 3) to provide an open-source code and working principles of these methods. The raw gyroscopic and accelerometer data were preprocessed. The seven methods included gyroscopic integration (GI), accelerometer inclination (AC), basic complementary filter (BCF), Kalman filter (KF), digital motion processor (DMP, a proprietary algorithm), Madgwick filter (MW), and Mahony filter (MH). An apparatus was designed to test nine conditions that computed angles for rotation about three axes (roll, pitch, yaw) and three movement speeds (50°/s, 150°/s, 300°/s). Each trial lasted 25 min. The root-mean-squared error (RMSE) between the gold-standard value measured from the apparatus' encoder and the value calculated from each of the seven methods was determined. For roll and pitch, all methods accurately quantified angles (RMSE < 6°) at all speeds. For yaw, all methods except AC and DMP displayed RMSE < 6° at all speeds. AC could not be used for yaw angle computation, and DMP displayed RMSE > 6°. Researchers can utilize appropriate methods based on their study's application.**

Index Terms—**Accelerometer, complementary filter, gyroscope, Kalman filter (KF), Madgwick filter (MW), Mahony filter (MH).**
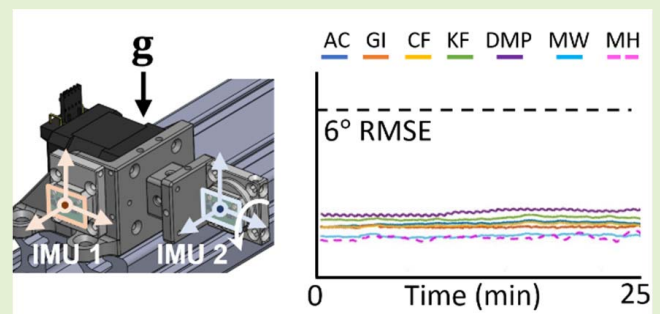
I. INTRODUCTION

RESEARCHERS have utilized compact, accurate, and affordable inertial measurement unit (IMU) sensors and developed algorithms to compute angular kinematics of one or more body segments in biomechanics and clinical applications [1], [2], [3], [4], [5], [6]. An IMU provides a 3-D vector representing the spatial orientation of the IMU. Thus, two IMUs can be used to measure the relative joint angle between two adjacent body segments by placing the IMUs on the body segments. This approach can be used for computing joint kinematics in biomechanical (e.g., knee flexion/extension angle during slow gait) [1], [2], [4], [7] and clinical applications (e.g., quantifying elbow joint kinematics during a clinical assessment of abnormal muscle behavior) [5], [6].

Previous studies have investigated the accuracy of high-performance IMUs under dynamic conditions encountered in human subject testing or test apparatuses. Seel *et al.* developed a method for quantifying human knee and ankle angles using 6-axis (i.e., 3-axis accelerometers and gyroscopes) IMUs while exploiting kinematic constraints. They validated the accuracy of IMUs via human subject testing using a motion capture system [1]. Favre *et al.* [8] introduced a method for measuring 3-D knee joint motion by aligning the reference frames of the two 6-axis IMUs of the thigh and shank segments. Lebel *et al.* validated multiple commercially available 9-axis IMUs under various testing conditions (e.g., various speeds and rotation axes) using an instrumented gimbal table [2], [9]. Ricci *et al.* [10] quantified the accuracy of commercial 9-axis IMUs under various speeds and rotation axes using a robotic arm. Bergamini *et al.* [11] proposed an adaptive Kalman filter (KF) method with a 6-axis IMU for quantifying the trunk angles of sprint runners. Caruso *et al.* [12], [13] utilized a magneto-IMU and presented a method for suboptimally tuning the input parameters for the IMU filter designs. However, these studies had some limitations such as the 1) dependence on magnetometers; 2) use of costly IMUs; 3) incomprehensive study design; and 4) lack of an open-source code for computing angles from IMU data.

First, some studies relied on IMUs with magnetometers (i.e., 9-axis IMUs) for computing the orientation of the IMUs [2], [9], [10], [12], [13]. While the magnetometers provide a measurement of absolute heading angle, the magnetometers are vulnerable to magnetic interferences caused by ferrous objects and electrical appliances (e.g., space heaters, metal filing cabinets), making magnetometers difficult to use in indoor environments [14], [15]. Although calibration procedures exist to compensate for the magnetic interferences [16], the procedures must be conducted periodically, making the device less practical in an indoor setting. Other studies have observed or noted the effect of magnetic disturbances in an indoor lab environment [9], [10], [17], [18]. However, a 6-axis IMU (i.e., 3-axis accelerometers and gyroscopes) does not use magnetometers to compute the orientation of the IMUs. While a 6-axis IMU cannot compute absolute yaw angle like the 9-axis IMU, the 6-axis IMU can still compute relative yaw angle which is still valuable in computing angular kinematics in biomechanics or clinical application [1], [2], [3], [4], [5], [6]. Thus, we utilized 6-axis IMUs in this study, since they are more robust, simpler, and equally effective choices than a 9-axis IMU for our application (i.e., indoor applications).

Second, some studies did not fully control the dynamic test conditions [1], [8]. Movement speeds, range of motion, rotation axes, and test duration are critical factors that can affect the accuracy and behavior of the IMUs [12], [13], [19], [20]. However, some studies only validated their methods using an arbitrarily defined movement speed and range of motion, or a very short test duration (<2 min) [9], [10], [11], [12].

Lastly, most of the studies did not provide an open-source code for researchers in other fields to easily implement these newly proposed methods [1], [8], [10], [11]. Researchers unfamiliar with control/estimation theory may have difficulty replicating and utilizing the newly proposed methods.

Thus, the goals of this study were to: 1) provide a low-cost and convenient IMU system utilizing two 6-axis IMUs without magnetometers for computing a relative angle between a stationary and a moving IMU; 2) validate various methods of computing 3-D absolute angles (roll, pitch, yaw) of an IMU following strict test conditions (i.e., long test duration, three movement speeds, three rotation axes); and 3) provide an open-source code and working principles of these methods to assist researchers unfamiliar with IMUs. The open-source code and data are at: https://github.com/ssong47/compute_relative_angle_between_two_IMUs, https://ieee-dataport.org/open-access/estimating-relative-angle-between-two-6-axis-inertial-measurement-units-imus

II. METHODS

A. Hardware and Test Apparatus

We developed an apparatus that performed programmable and repeatable rotations of IMUs and a gold standard measurement of the rotation angles (via an encoder) to validate the accuracy of our IMU system (see Fig. 1). Unlike other studies that validated their system through human subject testing by using optoelectronic motion capture data as the gold standard [1], [21], [22], our IMUs were mounted on this apparatus for
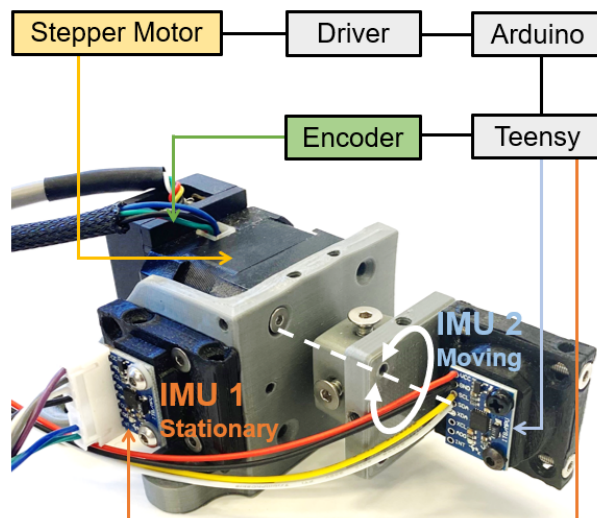

Fig. 1. Test apparatus for validating the IMU system. The setup shown here is configured for pitch rotation [see Fig. 2(b)].

testing to remove the errors due to motion artifacts from the attachment on a body segment. The apparatus consisted of two IMUs (MPU-6050; TDK-InvenSense, San Jose, CA), a stepper motor (NEMA 17, 42BYGH3401; Han Ding Motor, China) with an optical rotary incremental encoder (HEDS-9040#T00; Broadcom, USA) attached at the shaft, a stepper motor driver (A4988; Allegro Microsystems, Manchester, NH), and two microcontrollers (Teensy 3.6; PJRC, Sherwood, OR, and Arduino Uno; Arduino LLC, Italy). The MPU-6050 was chosen due to its compact size (smaller than a U.S. quarter), low power consumption, and low cost (~$US 4). An onboard proprietary digital motion processor (DMP[1]) algorithm was available for computing the 3-D orientation of each IMU, represented in quaternion form. Other system components are also economical and popular off-the-shelf choices (Arduino Uno ~$US 27, Teensy 3.6 ~$US 27, and NEMA 17 stepper motor ~$US 12).

The test apparatus used the Teensy 3.6 microcontroller to ensure a consistent sampling rate (100 Hz) for data collection and reliable transmission of data [i.e., timestamp, encoder angle, two IMUs' output data (quaternion from DMP, 3-D linear accelerations, and 3-D angular velocities in x, y, z axes)] to a PC. The I2C protocol ensured synchronous sampling of the data from the two IMUs since the sampling of each IMU was done simultaneously by using a shared clock signal. The Arduino Uno sent the PWM signals stepper motor driver to control the motor speed and range.

The encoder measurement served as the ground truth to evaluate the performance of the IMU system. The two IMUs, fixed on their custom mounts, were placed on different parts of the apparatus: IMU 1 on the stationary motor mount and IMU 2 on the rotating motor hub. The custom 3-D printed parts were designed to allow relative rotation about three orthogonal axes (i.e., yaw, pitch, roll) between the two IMUs (see Fig. 2). Depending on the axis of rotation of interest, the IMU mounts were configured differently (see Fig. 2). Note that the yaw rotation was configured such that the rotation axis was parallel
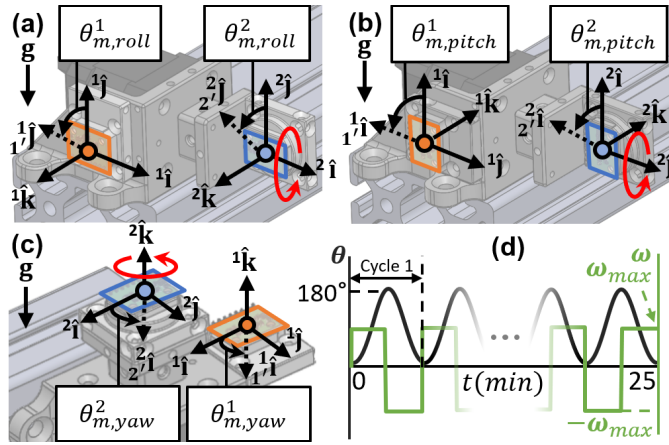
Fig. 2. Hardware and IMU setups of test configurations for three rotation axes [(a) roll, (b) pitch, and (c) yaw] and (d) θ, ω profiles for controlling the stepper motor. Orange and blue rectangles represent the stationary and moving IMUs. The red arrow indicates the rotation axis.

to the gravity vector [see Fig. 2(c)] to simulate a case in which gravity could no longer be a reference vector for the angle computation.

B. Testing Protocol

A total of nine test trials (three rotation axes × three movement speeds) with each trial length of 25 minutes and a range of motion of 180° was performed. 25 min was chosen as the test length since previous studies involving clinical research on neuro-rehabilitation lasted approximately 25 min for each test subject [3], [23], [24], [25]. The range of motion was set to 180° since most anatomical joints do not exceed 180°. Three speeds were chosen as the following: slow (max speed, $\omega_{\max} = 50°/s$), medium ($\omega_{\max} = 150°/s$), and fast ($\omega_{\max} = 300°/s$) to simulate the kinematics of an elbow or wrist joint during a clinical assessment (20°/s–180°/s) [6], [26], [27], [28] as well as of a knee joint during the walking phases at a slow cadence (23°/s–250°/s) [29], [30], [31]. Similar movement speeds and rotation axes were used in other studies [9], [10]. The motor driver controlled the angular position and speed profile shown in Fig. 2(d).

Before each trial, the two IMUs were removed from the test apparatus and calibrated to remove any unwanted bias and scaling factor of the IMU readings. The calibration protocol followed the common six-position static test mentioned in [32], [33]. At the beginning of each trial, the principal axes of IMU 1 and 2 were aligned parallel to each other. Also, the encoder angle was set to zero. During each trial, the two IMU and encoder readings were sampled at 100 Hz.

C. Methods for Computing Relative Angles

Seven methods of computing relative angles about a single rotation axis were investigated. These included the DMP method, gyroscopic integration (GI), accelerometer inclination (AC), basic complementary filter (BCF), KF, Madgwick filter (MW), and Mahony filter (MH). The mathematical formulas, as well as the pros and cons of each method, are discussed here.

The testbed setup had one stationary IMU (IMU 1) and one rotating IMU (IMU 2), where two IMUs' local frames were always aligned by construction in all test configurations (see Fig. 2). The relative angle ($\theta_{m,\hat{\mathbf{u}}}$) was defined as how much the moving IMU (IMU 2) rotated relative to stationary IMU 1 around the testing axis, where $m \in$ {GI, AC, BCF, KF, MW, MH, DMP}, indicating the algorithm being used for calculation. For pitch and roll [see Fig. 2(a) and (b)], $\theta_{m,\hat{\mathbf{u}}}$ was obtained by calculating the inclination angle of each IMU relative to the horizontal plane and then subtracting the two inclination angles (1). For yaw [see Fig. 2(c)], since the motion is in the horizontal plane, instead of inclination angle, the change in the angle with respect to the initial angle of each IMU was first calculated and then subtracted to obtain $\theta_{m,\hat{\mathbf{u}}}$

$$\theta_{m,\hat{\mathbf{u}}} = \theta_{m,\hat{\mathbf{u}}}^2 - \theta_{m,\hat{\mathbf{u}}}^1 \tag{1}$$

where the superscript denoted the IMU of interest and $\hat{\mathbf{u}}$ denoted the rotation axis ($\hat{\mathbf{u}} \in \{\hat{\mathbf{i}}(\text{roll}), \hat{\mathbf{j}}(\text{pitch}), \hat{\mathbf{k}}(\text{yaw})\}$). In the case of pitch and roll rotations, to obtain the inclination angle of each IMU, when using quaternion-based methods (i.e., DMP, MW, and MH), $\theta_{m,\hat{\mathbf{u}}}^1$ and $\theta_{m,\hat{\mathbf{u}}}^2$ were obtained by converting the quaternion of each IMU into intrinsic Euler angles ("ZYX" order) and then selecting the Euler angle of interest (roll, pitch, or yaw angles), whereas when using axis-angle-based methods (i.e., GI, AC, BCF, and KF), $\theta_{m,\hat{\mathbf{u}}}^1$ and $\theta_{m,\hat{\mathbf{u}}}^2$ were obtained directly around the rotation axis being tested.

Each IMU outputted 3-D gyroscope values, accelerometer values, and a unit quaternion vector (**q**) from DMP. **q** contained four values (a, b, c, d, where a defined the amount of rotation and b, c, d defined the axis of rotation in the 3-D Cartesian space) to quantify rotation relative to the reference frame of the IMU ($\hat{\mathbf{i}}, \hat{\mathbf{j}}, \hat{\mathbf{k}}$) (2) [34]

$$_{A'}^{A}\mathbf{q} = a + b_{A'}^{A}\hat{\mathbf{i}} + c_{A'}^{A}\hat{\mathbf{j}} + d_{A'}^{A}\hat{\mathbf{k}}. \tag{2}$$

This article follows the vector notation recommended by Arkfen and Weber [35]: bold serif for vectors (e.g., **q**), bold lowercase letter with circumflex for unit vectors of a reference frame (e.g., $\hat{\mathbf{i}}$), and left superscript and subscript indicating the reference frame $\{A\}$ and inertial frame $\{A'\}$, respectively (e.g., $_{A'}^{A}\mathbf{q}$). The frames of IMU 1 and 2 are defined as $i \in \{1, 2\}$

$$\begin{bmatrix} \theta_x \\ \theta_y \\ \theta_z \end{bmatrix} = \begin{bmatrix} a\tan2\left(2\left(bc + ad\right), a^2 + b^2 - c^2 - d^2\right) \\ a\sin\left(-2\left(bd - ac\right)\right) \\ a\tan2\left(2\left(cd + ab\right), a^2 - b^2 - c^2 + d^2\right) \end{bmatrix}. \tag{3}$$

Each quaternion can be converted into Euler angles using MATLAB scripting (3) [34], [36].

1) Digital Motion Processor (DMP): The DMP method, developed commercially by TDK—InvenSense, is a proprietary algorithm that fuses the 3-D acceleration and 3-D gyroscope data and outputs the 3-D orientation of an IMU in quaternion form [37]. While the computation of a quaternion from an IMU is proprietary, the computation of relative angles of the two IMUs is not. These quaternions ($_{1'}^{1}\mathbf{q}$ or $_{2'}^{2}\mathbf{q}$) were converted to Euler angles ($\theta_z^i, \theta_y^i, \theta_x^i$) using (3). The angle of interest from the Euler angles was used to compute the absolute angle of

IMU 1 or IMU 2, and (1) was used to compute the relative angle between IMU 1 and 2.

Algorithm 1 Digital Motion Processor$^{\text{TM}}$ (DMP)

$$quaternion^i_{t,}\mathbf{q} \rightarrow Euler\ angles\ \left\{\theta^i_z, \theta^i_y, \theta^i_x\right\}$$
$$\theta^i_{DMP} = \theta^i_x\ (\text{roll})\ \theta^i_{DMP} = \theta^i_y\ (\text{pitch})\ \theta^i_{DMP} = \theta^i_z\ (\text{yaw})$$

---

2) Gyroscopic Integration (GI): GI integrates gyroscopic data at each sampled time [38]. Since integrating raw gyroscopic data would cause drift due to inherent noise and bias instability [38], the raw gyroscopic signals were preprocessed using a fourth-order Butterworth high-pass filter with a cutoff frequency of 0.07 Hz. A similar approach was done by other studies [39], [40], [41]. While the most optimal cut-off frequency may be unique for different movement speeds, we wanted to investigate if a single filter design was robust enough to accommodate a range of movement speeds. The cutoff frequency was chosen low enough to ensure that signals related to the motion at all speeds were collected but high enough to reduce the effect of the long-term effect of drifting of gyroscopes [42], [43]. The lowest frequency of motion was 0.139 Hz ($=(1/360°/(50°/s))$), which was determined by calculating the time it takes to complete one cycle of motion ($\theta_c = 360° = 2 \times 180°$, since the motor rotated 180° in one direction and came back) at the slowest rotational speed ($\omega = 50°/s$). Thus, a cut-off frequency of 0.07 Hz (<0.139 Hz) ensured that signals related to motions at all speeds were collected with minimal unwanted gyroscopic drift.

Algorithm 2 Gyroscopic Integration (GI) [38]

$$\theta^i_{t,GI_{\hat{\mathbf{u}}}} = \omega^i_{hf,t_{\hat{\mathbf{u}}}}\Delta t + \theta^i_{t-1,GI_{\hat{\mathbf{u}}}}$$

For IMU i, $\theta^i_{t,GI_{\hat{\mathbf{u}}}}, \theta^i_{t-1,GI_{\hat{\mathbf{u}}}}, \omega^i_{hf,t_{\hat{\mathbf{u}}}}$, and Δt were the angles from GI at the tth data point about the rotation axis $\hat{\mathbf{u}}$, the angle from GI at $t-1$th data point about $\hat{\mathbf{u}}$, the high-pass-filtered angular velocity gyro at the tth data point data about $\hat{\mathbf{u}}$, and sampling time ($=0.01$ s), respectively. The advantages of GI are easy implementation and low computational cost [44], [45].

3) Accelerometer Inclination (AC): AC solely uses the accelerometer data and computed the inclination angle using the gravity vector and its projection on the axes of the accelerometer [45]. The accelerometer data were preprocessed using a low-pass fourth-order Butterworth filter with a cut-off frequency of 4 Hz to fully capture the signals with a frequency (4 Hz > 0.833 Hz $= (1/2(180°)/(300°/s))$) of the motion even at the fastest speeds [46]. A single cut-off frequency was chosen for similar reasons as the high-pass filter mentioned above. The yaw angle cannot be obtained for this method since the yaw has a rotation axis parallel to the gravity vector $\mathbf{g}$, making the projections of $\mathbf{g}$ on the accelerometer to be zero. $\theta^i_{t,AI_{\hat{\mathbf{i}}}}, \theta^i_{t,AI_{\hat{\mathbf{j}}}}, A^i_{t,\hat{\mathbf{i}}}, A^i_{t,\hat{\mathbf{j}}}$, and $A^i_{t,\hat{\mathbf{k}}}$ were angles from AC about $\hat{\mathbf{i}}$ (roll), $\hat{\mathbf{j}}$ (pitch) axes and low-pass-filtered acceleration readings about $\hat{\mathbf{i}}, \hat{\mathbf{j}}, \hat{\mathbf{k}}$ axes, respectively.

The advantages of the AC method were drift-free computation of angle, simple implementation, and low computational

---

Algorithm 3 Accelerometer Inclination (AC) [45]

$$\theta^i_{t,AC_{\hat{\mathbf{i}}}} = \tan^{-1}\left(\frac{A^{i^2}_{t,\hat{\mathbf{i}}}}{\sqrt{A^{i^2}_{t,\hat{\mathbf{j}}}+A^{i^2}_{t,\hat{\mathbf{k}}}}}\right),\ \theta^i_{t,AC_{\hat{\mathbf{j}}}} = \tan^{-1}\left(\frac{A^{i^2}_{t,\hat{\mathbf{j}}}}{\sqrt{A^{i^2}_{t,\hat{\mathbf{i}}}+A^{i^2}_{t,\hat{\mathbf{k}}}}}\right)$$

---

cost [45]. The disadvantages of AC were the high sensitivity to motion/dynamic accelerations (i.e., acceleration not due to gravity) and the inability to compute yaw angles [42], [47]. The angle from AC contained high-frequency noise, which could be mitigated using low-pass-filtered accelerometer readings [42], [43], [45].

4) Basic Complementary Filter (BCF): The BCF method combines two filters: a low-pass filter for accelerometer signals and a high-pass filter for gyroscopic signals [42], [48], [49]. The two filters are complements of each other, so the summation of the two filtered signals results in a gain of one. The advantages of BCF were smooth, accurate, and drift-free angle computation that combined the benefits of GI and AC while maintaining low computational costs. Also, the tuning process was simple due to only one tunable parameter (γ) that determined the cut-off frequency of the low- and high-pass filters [50]. γ was determined by the sampling rate (100 Hz) and time constant (τ) of the system. τ was determined by the gyroscope's drift rate (ω_{drift}) and the tolerable error (e_{tol}). For our test setup and data, the measured ω_{drift} of MPU 6050 was 0.25°/s. For e_{tol}, 3° was chosen to ensure the tolerable error was at least half of the threshold for allowable error in biomechanics studies (6°) [1], [2], [21], [51]. Thus, τ was set to 0.083 s^{-1} ($=0.25/3$), which meant that BCF would perform GI for signals with frequencies larger than 0.083 Hz, while the accelerometer readings would be filtered out. For signals with frequencies smaller than 0.083 Hz, BCF would rely on AC angles, while the gyroscopic readings would be ignored. Thus, $\gamma = 0.11 \cong 1 - (0.083/0.083 + 0.01)$ was chosen. For pitch and roll angles, the raw gyroscopic signals were not filtered for BCF, since the BCF algorithm already corrects for the gyroscopic drift using accelerometer values. Having extra high-pass filters is redundant and may increase unwanted filter artifacts (e.g., delays) that have an adverse effect on real-time applications. For yaw angles, a small value of γ ($= 10^{-7}$) was used to put less weight in the accelerometer readings since the accelerometer did not provide any relevant information regarding the yaw rotation. Also, the gyroscopic signals were high-pass filtered through the same filter settings used in GI, since the accelerometer readings could not properly correct for the gyroscopic drift.

Algorithm 4 Basic Complementary Filter (BCF) [42], [48], [49]

$$\tau = \frac{\omega_{drift}}{e_{tol}}\quad\gamma = 1 - \frac{\tau}{\tau+\Delta t}$$
$$\theta^i_{t,CF_{\hat{\mathbf{u}}}} = (1-\gamma)(\omega^i_t \Delta t + \theta^i_{t-1,CF}) + \gamma\,\theta^i_{t,AC_{\hat{\mathbf{u}}}}$$

$\theta^i_{t,CF_{\hat{\mathbf{u}}}}$ was the BCF angle about the $\hat{\mathbf{u}}$ axis.

5) Kalman Filter (KF): KF computes angle via an optimal state estimator algorithm, consisting of prediction and update

steps [16], [42], [52]. The prediction step involves estimation of the *a priori* (i.e., at current time step without any observation information from current time step) system states ($_{[2\times1]}\hat{\mathbf{x}}^i_{t|t-1}$) and *a priori* error covariance ($_{[2\times2]}\boldsymbol{P}^i_{t|t-1}$) from state-space system dynamics and inputs (u^i_t). The updating step combines the current observation information ($_{[2\times1]}\tilde{\mathbf{y}}^i_t$) to improve the state estimates, obtaining *a posteriori* (i.e., current time step with observation information) state estimates ($_{[2\times1]}\hat{\mathbf{x}}^i_{t|t}$), as well as updating the *a posteriori* estimate covariance ($_{[2\times1]}\boldsymbol{P}^i_{t|t}$). In this step, the Kalman gain ($_{[2\times1]}\boldsymbol{K}^i_t$) was found by optimally balancing the error of sensor measurements and state estimates, given the initial estimates of the states ($_{[2\times1]}\hat{\mathbf{x}}^i_0$) and process covariance matrix ($_{[2\times2]}\boldsymbol{Q}$) and measurement noise covariance matrix ($_{[2\times2]}\boldsymbol{R}$). Note that the KF in our setup did not calculate the full 3-D orientation of the IMU, but only computed the angle of interest about the rotation axis at a time (i.e., roll, pitch, yaw). For example, the angle about the roll axis ($\theta^i_{t,\mathrm{KF}_{\hat{\mathbf{x}}}}$) was computed using $\omega^i_{t,\hat{\mathbf{x}}}$ and $\theta^i_{t,\mathrm{AC}_{\hat{\mathbf{x}}}}$.

Algorithm 5 Kalman Filter (KF) [16], [42], [52]

I. State Space Definition

$$\hat{\mathbf{x}}^i_t = \begin{bmatrix} \theta^i_{t,KF_{\hat{\mathbf{u}}}} \\ b^i_{t,gyro} \end{bmatrix} \quad u^i_t = \omega^i_{t,\hat{\mathbf{u}}}$$

$\hat{\mathbf{u}} = \{\hat{\mathbf{x}}, \hat{\mathbf{y}}, \hat{\mathbf{z}}\}$ for $\{roll, pitch, yaw\}$, respectively.

$$\hat{\mathbf{x}}^i_0 = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad P_0 = \begin{bmatrix} 10^6 & 0 \\ 0 & 10^6 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & -\Delta t \\ 0 & 1 \end{bmatrix} \quad B = \begin{bmatrix} \Delta t & 0 \\ 0 & 0 \end{bmatrix} \quad C = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}$$

II. Prediction Step

$$P^i_{t|t-1} = A P^i_{t-1|t-1} A^T + Q$$
$$\hat{\mathbf{x}}^i_{t|t-1} = A\hat{\mathbf{x}}^i_{t-1|t-1} + B\mathbf{u}^i_t \quad \hat{\mathbf{y}}^i_t = C\hat{\mathbf{x}}^i_{t|t-1}$$

III. Update Step

$$K^i_t = P^i_{t|t-1}C^T \left(R + C P^i_t C^T\right)^{-1}$$
$$\tilde{\mathbf{y}}^i_t = [\theta^i_{t,AC_{\hat{\mathbf{u}}}} \, \mathbf{0}]^T - \hat{\mathbf{y}}^i_t \quad \hat{\mathbf{x}}^i_{t|t} = \hat{\mathbf{x}}^i_{t|t-1} + K^i_t \tilde{\mathbf{y}}^i_t$$
$$P^i_{t|t} = \left(I - K^i_t C\right) P^i_{t|t-1}$$

$b^i_{t,\mathrm{gyro}}$ was the bias of the gyroscope. Note that the initial error covariance ($\boldsymbol{P}_0$) was set to a relatively high value in our study to ensure fast convergence [53].

$\boldsymbol{Q}$ was a diagonal matrix since we assumed that the variance of accelerometer noise and gyroscope noise was independent. $\boldsymbol{Q}_{22}$ was equivalent to the variance (σ^2_ω) of raw gyroscopic noise. Assuming all zero mean gyroscopic noise ($\bar{p}_\omega = 0$), $\sigma^2_\omega = p^2_{\omega,\mathrm{RMS}}$, where $p_{\omega,\mathrm{RMS}}$ was the rms noise of our gyroscope. $p_{\omega,\mathrm{RMS}} = p_{\omega,\mathrm{nspd}}\sqrt{\Delta t^{-1}}$, where Δt^{-1} was our sampling rate ($=100$ Hz), and $p_{\omega,\mathrm{nspd}}$ was the noise spectral density ($=0.005°/s/\sqrt{\mathrm{Hz}}$) of MPU6050 [37]. $p_{\omega,\mathrm{RMS}}$ was computed using Δt^{-1} and $p_{\omega,\mathrm{nspd}}$ rather than using the reported total rms noise value, since the gyroscopic noise level depends not only on the performance of the sensor, but also on the sampling rate [54]. Thus, $\boldsymbol{Q}_{22} = 2.5 \times 10^{-3}$. Like other studies [42], $\boldsymbol{Q}_{11}$ was set to 1.0×10^{-3} to maintain

similar order of magnitude as $\boldsymbol{Q}_{22}$. $\boldsymbol{R}(=\boldsymbol{I}_2 \times 3.76°)$ was determined by empirically measuring the variance of the AC angle's noise. For pitch and roll, the raw gyroscopic signals were not filtered for KF, since the KF algorithm already tries to remove the gyroscopic bias through estimation of the bias within the state space model. For yaw rotations, $\boldsymbol{R}$ was set to a matrix with large numbers ($=\boldsymbol{I}_2 \times 10^8$) since the accelerometer measurements were not reliable for yaw. Like BCF, the gyroscopic signals were high-pass filtered through the same filter used in GI, since the accelerometer readings could not properly correct for the gyroscopic drift. The benefit of KF is that it optimally computes the gain that balances the inaccuracy of the measurements and the model at every time step.

6) Madgwick Filter (MW): The MW is a sensor fusion algorithm that combines the gyroscope and accelerometer data in a quaternion representation to analytically derive and optimize a gradient-descent (GD) algorithm to compute a quaternion derivative of the IMU [47], [55]. First, the MW estimates the IMU (i) quaternion derivative ($^i_{i'}\dot{\mathbf{q}}_{\omega,t}$) using the vector ($^i\boldsymbol{\omega}_t$) containing gyroscopic measurement ($^i\omega_{x,t}$, $^i\omega_{y,t}$, or $^i\omega_{z,t}$) and previous estimate of orientation ($^i_{i'}\mathbf{q}_{\mathrm{est},t-1}$). Only one gyroscopic signal of $^i\boldsymbol{\omega}_t$ was nonzero, since the IMU is rotated about a single axis. To minimize the drift from this gyroscopic measurement, a corrective quaternion derivative term ($^i_{i'}\dot{\mathbf{q}}_{\epsilon,t}$) is computed by estimating the direction of the gravitational vector $^i\mathbf{g}_{\mathrm{est}}$ using the GD-based algorithm on the accelerometer data ($^i_{i'}\mathbf{a}_t$). Note that we assumed that the accelerometer mostly measures the direction of $^i\mathbf{g}_{\mathrm{est}}$. In the GD, objective function f is minimized by using the Jacobian matrix (J) of f which computed the error direction ($(\nabla f / |\nabla f|)$) on the solution surface. A corrected quaternion derivative term ($^i_{i'}\dot{\mathbf{q}}_{\mathrm{est},t}$) is computed by fusing the quaternion derivative estimations from the gyroscope ($^i_{i'}\dot{\mathbf{q}}_{\omega,t}$) and the accelerometer ($^i_{i'}\dot{\mathbf{q}}_{\epsilon,t}$) using an adjustable gain β. β represents all mean zero gyroscope measurement errors ($\tilde{\omega}_\beta$), expressed as the quaternion derivative form. $\tilde{\omega}_\beta (= 0.05°/s)$ is dependent on the sensor's gyroscopic performance. The value of $\beta = 0.043$ was chosen by using the equation ($\beta = \sqrt{3}/4\tilde{\omega}_\beta$) suggested by Madgwick *et al.* [47]. For yaw rotations, $\beta = 0$ since the accelerometer readings did not describe the movement of the IMU properly when the rotation axis is parallel to $\mathbf{g}$. The IMU's orientation estimation ($^i_{i'}\mathbf{q}_{\mathrm{est},t}$) was computed by integrating the corrected quaternion derivative ($^i_{i'}\dot{\mathbf{q}}_{\mathrm{est},t}$). Similar procedure from DMP involving conversion from quaternion to Euler angles was used in MW to compute the relative angles between IMU 1 and 2.

For pitch and roll, the raw gyroscopic signals ($^i\omega_{x,t}$, $^i\omega_{y,t}$) were used without a high-pass filter since the gyroscopic drifting behavior was corrected by the accelerometer data. An additional high-pass filter was redundant and added delays. However, for yaw, high-pass-filtered gyroscopic signals ($^i\omega_{\mathrm{hf},z,t}$) were used, since MW cannot rely on the accelerometer data to compensate for gyroscopic bias.

The advantages of MW are its low computational load, low sampling rate requirement, and simple tuning method (only one adjustable parameter—β). Since MW uses a quaternion representation, the filter is not subject to gimbal lock problems associated with Euler angle representations. Note that

a potential issue with MW is the unpredictable convergence behavior and a more recent version of this algorithm with enhanced robustness was reported in [56].

Algorithm 6 Madgwick Filter (MW) [47], [55]

$$^i\boldsymbol{\omega}_t = \begin{bmatrix} ^i\omega_{x,t} & 0 & 0 \end{bmatrix} \text{ for roll}$$
$$^i\boldsymbol{\omega}_t = \begin{bmatrix} 0 & ^i\omega_{y,t} & 0 \end{bmatrix} \text{ for pitch}$$
$$^i\boldsymbol{\omega}_t = \begin{bmatrix} 0 & 0 & ^i\omega_{hf,z,t} \end{bmatrix} \text{ for yaw}$$
$$^i\boldsymbol{\omega}_t^q = \begin{bmatrix} 0 & ^i\boldsymbol{\omega}_t \end{bmatrix}$$
$$^i\mathbf{a}_t = \begin{bmatrix} 0 & ^ia_x & ^ia_y & ^ia_z \end{bmatrix}$$
$$^i\mathbf{g}_{est} = \begin{bmatrix} 0 & 0 & 0 & 9.81 m/s^2 \end{bmatrix}$$
$$^i_{i'}\dot{\mathbf{q}}_{\omega,t} = \tfrac{1}{2}^i_{i'}\mathbf{q}_{est,t-1} \otimes {}^i\boldsymbol{\omega}_t^q$$
$$\text{minimize} f\left(^i_{i'}\mathbf{q}_{est,t-1}, {}^i\mathbf{a}_t \right)$$
$$f\left(^i_{i'}\mathbf{q}_{est,t-1}, {}^i\mathbf{a}_t \right) = {}^i_{i'}\mathbf{q}_{est,t-1}^* \otimes {}^i\mathbf{g}_{est} \otimes {}^i_{i'}\mathbf{q}_{est,t-1} - {}^i\mathbf{a}_t$$
$$^i_{i'}\mathbf{q}_{est,t-1}^* = {}^{i'}_i\mathbf{q}_{est,t-1}$$
$$\nabla f = J\left(^i_{i'}\mathbf{q}_{est,t-1} \right) f\left(^i_{i'}\mathbf{q}_{est,t-1}, {}^i\mathbf{a}_t \right)$$
$$^i_{i'}\dot{\mathbf{q}}_{\epsilon,t} = \frac{\nabla f}{|\nabla f|}$$
$$^i_{i'}\dot{\mathbf{q}}_{est,t} = {}^i_{i'}\dot{\mathbf{q}}_{\omega,t} - \beta^i_{i'}\dot{\mathbf{q}}_{\epsilon,t} \quad \beta = \sqrt{3/4}\tilde{\omega}_\beta$$
$$^i_{i'}\mathbf{q}_{est,t} = {}^i_{i'}\mathbf{q}_{est,t-1} + {}^i_{i'}\dot{\mathbf{q}}_{est,t}\Delta t$$

$\otimes, i$ were the quaternion product to define sequential rotations [47] and IMU of interest ($i \in \{1, 2\}$), respectively.

7) Mahony Filter (MH): The MH is another variation of a BCF that corrects the gyroscopic measurement by a proportional-integral (PI) compensator [55], [57]. First, MH computes the error ($^i\mathbf{e}_t$) between the estimated gravity vector ($^i\mathbf{a}_{est,t}$) and accelerometer readings ($^i\mathbf{a}_t$) represented by the cross product between $^i\mathbf{a}_t$ and $^i\mathbf{a}_{est,t}$. Then, the gyroscopic vector ($^i\boldsymbol{\omega}_t^q$) containing gyroscopic measurements ($^i\omega_{x,t}, {}^i\omega_{y,t},$ or $^i\omega_{z,t}$) is combined with $^i\mathbf{e}_t$ adjusted by PI gains (K_p, K_i) to obtain a corrected estimate of the gyroscopic vector ($^i\boldsymbol{\omega}_{est,t}^q$). Afterward, the quaternion derivative ($^i_{i'}\dot{\mathbf{q}}_{est,t}$) is obtained from the previous estimate of quaternion ($^i_{i'}\mathbf{q}_{est,t-1}$) and $^i\boldsymbol{\omega}_{est,t}^q$. Like MW, only one gyroscopic signal of $^i\boldsymbol{\omega}_t$ is nonzero. The IMU's orientation quaternion ($^i_{i'}\mathbf{q}_{est,t}$) is obtained by integrating the quaternion derivative ($^i_{i'}\dot{\mathbf{q}}_{est,t}$). A similar procedure as DMP, that is, conversion from quaternion to Euler angles, was used in MH to compute the relative angles between IMUs 1 and 2. The filter gains (K_p, K_i) were initially set to their default values (1.0, 0.3) and then fine-tuned heuristically (5.0, 0.3) for pitch and roll rotations. For yaw rotations, $K_p = K_i = 0$ due to similar reasons as MW. Like MW, raw gyroscopic signals were used for pitch and roll, while high-pass-filtered gyroscopic signals were used for yaw. The MH has been reported to be an efficient and effective solution by other studies [55], [58].

D. Data Processing

The performance of each method was evaluated using root-mean-squared error (RMSE) between the encoder and the seven computed methods for evaluating the accuracy of
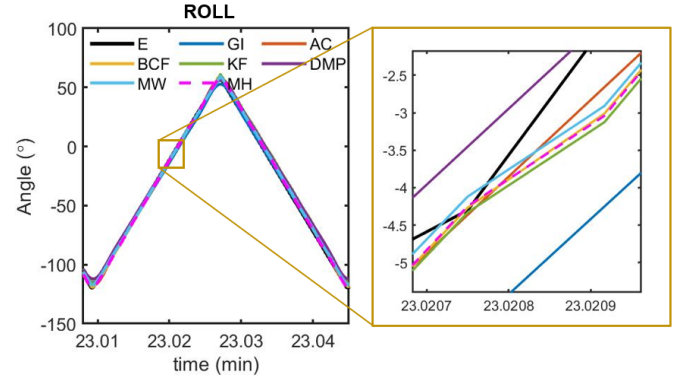

Algorithm 7 Mahony Filter (MH) [55], [57]

$$^i\mathbf{a}_{est,t} = {}^i_{i'}\mathbf{q}_{est,t-1}^* \otimes {}^i\mathbf{g}_{est} \otimes {}^i_{i'}\mathbf{q}_{est,t-1}$$
$$^i\mathbf{e}_t = {}^i\mathbf{a}_t \times {}^i\mathbf{a}_{est,t}$$
$$^i\boldsymbol{\omega}_{est,t}^q = {}^i\boldsymbol{\omega}_t^q + \begin{bmatrix} 0 & K_p {}^i\mathbf{e}_t \end{bmatrix} - \begin{bmatrix} 0 & K_i {}^i\mathbf{e}_t \Delta t \end{bmatrix}$$
$$^i_{i'}\dot{\mathbf{q}}_{est,t} = \tfrac{1}{2}^i_{i'}\mathbf{q}_{est,t-1} \otimes {}^i\boldsymbol{\omega}_{est,t}^q$$
$$^i_{i'}\mathbf{q}_{est,t} = {}^i_{i'}\mathbf{q}_{est,t-1} + {}^i_{i'}\dot{\mathbf{q}}_{est,t}\Delta t$$

Fig. 3. Angle across time near the end of a 25-min trial at medium speeds (150°/s) for roll rotations (left). A magnified plot was also shown (right). All seven methods computed accurate angles following the encoder (E) angles. Trials for other speeds and rotations displayed similar behavior, with the exception of DMP during yaw trials.

the computed angle's magnitude. For RMSE computation, a window size of one minute, or 6000 sample points (=60 s × 100 points/s) were chosen for two reasons: 1) to better visualize the trends of RMSE changes and 2) since each test trial from other research studies lasted approximately 1 min [11], [59], [60]. RMSE of 6° was chosen as the maximum acceptable limit for biomechanical research studies [1], [2], [21], [51]. All RMSE data were truncated such that the start (0–1 min) and end (24–25 min) data were removed. The artifacts were due to the preprocessing of gyroscopic and accelerometer signals. For each speed, the average ($\overline{\text{RMSE}}$) and standard error (σ_{RMSE}) RMSE were computed for the truncated data. In addition to the RMSE data, the average computational time (t_{comp}) required to execute a single iteration for each algorithm was calculated for all seven methods. All data were processed using MATLAB R2020a.

III. RESULT

A. Roll and Pitch

All methods exhibited high accuracy (RMSE < 6°) at all movement speeds and no signs of drift across the entire duration (see Figs. 3–5). Also, the RMSE of these methods was generally proportional to speed (see Fig. 4). While the $\overline{\text{RMSE}}$ differences between these methods were less than 3°, the methods displayed differences in terms of consistency of RMSE trends across time and movement speeds.

The four sensor fusion algorithms (BCF, KF, MW, MH) displayed the lowest $\overline{\text{RMSE}}$ across all speeds for all rotation axes (see Figs. 4 and 5). In addition, the RMSE remained consistent across time with only minor fluctuations (~0.5°).
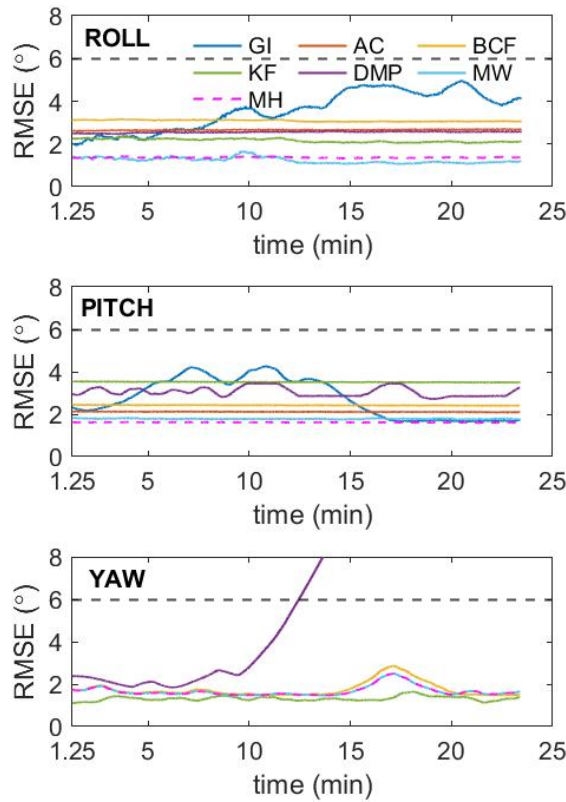
Fig. 4. RMSE across time for trials at medium speeds (150°/s) for roll (top), pitch (middle), and yaw (bottom). For yaw trials, MW and MH had identical RMSE values. Tolerable RMSE (6°) is shown as a black dashed horizontal line.
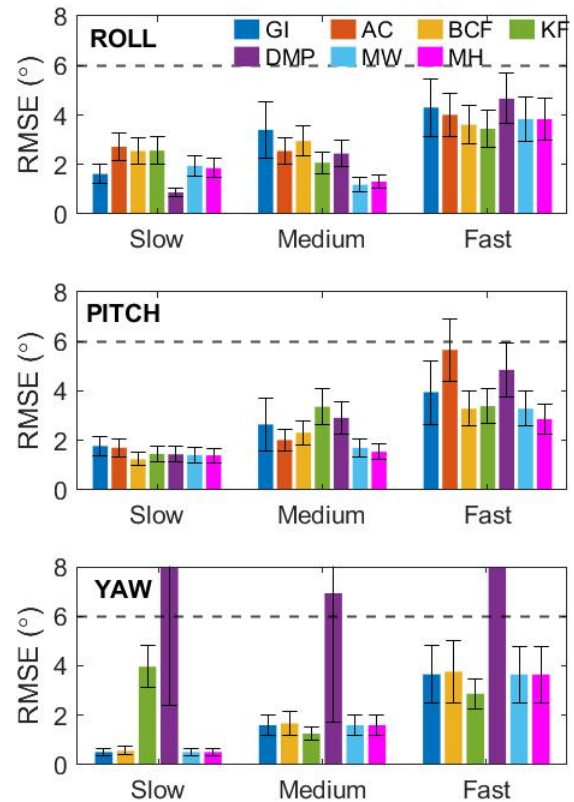

Fig. 5. Average RMSE with standard error of each algorithm for each and all speeds (slow—50°/s, medium—150°/s, fast—300°/s) for roll (top), pitch (middle), and yaw (bottom). For yaw trials, the average RMSE of MW and MH were identical. Tolerable RMSE (6°) is shown as a black dashed horizontal line.
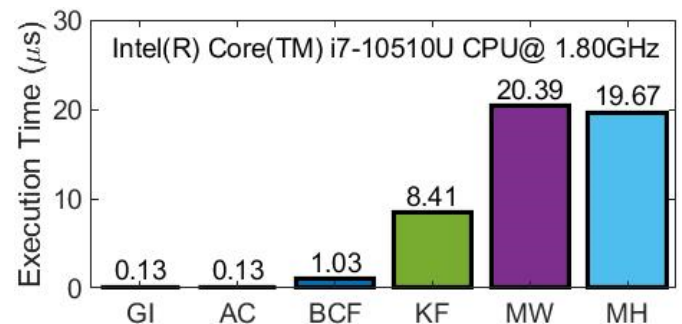

Fig. 6. Average computational execution time needed for a single update step by each method across all speeds and rotations. Note that DMP was unknown since DMP was computed using a proprietary algorithm.

The $\overline{\text{RMSE}}$ was proportional to movement speeds, since the $\overline{\text{RMSE}}$ was the highest (3°–5°) at fast speed and the lowest (1°–3°) at slow speed.

AC, GI, and DMP displayed slightly larger errors than the previously mentioned four methods. The magnitude of $\overline{\text{RMSE}}$ was higher by 1°–3° for all speeds. While GI had a relatively large error and fluctuations compared to others, GI's RMSE did not drift and remained consistently below the 6° threshold.

B. Yaw

Five of the methods (GI, BCF, KF, MW, MH) displayed high accuracy ($\overline{\text{RMSE}} < 6°$) and consistency ($\sigma_{\text{RMSE}} < 3°$) without any drifting behavior for yaw rotations. Like pitch and roll, the RMSE of these methods was generally proportional to the movement speed with a few exceptions (e.g., for yaw, $\overline{\text{RMSE}}_{\text{CF}}$ at slow $> \overline{\text{RMSE}}_{\text{CF}}$ at medium speed). GI exhibited the highest error among the five methods, followed by KF and BCF. MW and MH showed the lowest error. Note that MW and MH had identical RMSE values (see Figs. 4 and 5).

The other two methods (AC, DMP) either could not be computed at all (AC) or displayed high errors (DMP). DMP drifted, resulting in high $\overline{\text{RMSE}}$ and high σ_{RMSE}.

C. Computational Execution Time

The average computational execution times required for a single update step were lowest for the GI, AC, followed by BCF, KF, MH, and MW (see Fig. 6). The execution time for BCF was marginally higher ($\times 8$) than GI and AC. For KF, the execution time was higher than GI and AC by $\sim 65\times$. The execution times for MW and MH were significantly higher than GI, and AC by a factor of almost $160\times$. The computational time for DMP could not be calculated since DMP was an onboard proprietary algorithm.

IV. DISCUSSION

The purpose of this study was to systematically validate seven computational methods (GI, AC, DMP, BCF, KF, MW, MH) to the gold standard (encoder) using two low-cost 6-axis IMUs without magnetometers. An explanation of the inner

workings and open-source code of the seven methods were presented to assist newcomers to the field of IMUs. A custom test apparatus and strict test protocols were used to control for critical factors affecting IMU accuracies, such as movement speeds, range of motion, rotation axes, and sufficiently long test durations.

The result of this study suggests that most methods (all methods for pitch/roll. GI, BCF, KF, MW, MH for yaw) using 6-axis IMUs can be used in a controlled indoor environment to accurately quantify angles for various movement speeds (50°/s–300°/s) and test durations of 25 min. The RMSE for these methods were all below the maximum acceptable RMSE for biomechanical study (=6°). The RMSE for our methods was generally proportional to movement speed, which was also observed in other studies [9], [10], [12], [19]. Noise introduced from the larger accelerations at faster speeds may be the cause for the slightly increased error. For pitch and roll rotations, all methods did not require retuning of filter parameters for this study's movement speeds, illustrating the method's robustness to the given movement speeds.

Adjustments in the filter parameter and preprocessing of gyroscopic signals had to be made for yaw trials. Filter parameters for BCF, KF, MW, and MH had to be changed to rely only on gyroscopic reading due to the absence of a magnetometer and unusable accelerometer readings for yaw rotations. Some methods could not be used (AC) or were inaccurate due to drift (DMP). Also, high-pass filtering of raw gyroscopic signals was necessary to remove the gyroscopic bias since accelerometer signals could not correct for the drifting of gyroscopic signals.

A. Roll and Pitch

BCF, KF, MW, and MH demonstrated the best performance (i.e., lowest RMSE) and consistency (i.e., lowest σ_{RMSE}), compensating for the weaknesses of each sensor to accurately compute the angles (Figs. 4 and 5, top and mid). While the sensor fusion algorithm innerworkings differed greatly for each method, the accuracy and consistency of each method were similar in our study. With carefully tuned filter parameters, relatively simple sensor fusion algorithms, such as BCF and KF performed as well as more advanced algorithms, such as MW and MH. BCF, KF, MW, and MH were easy to tune and implement since they required only one or two tuning parameters. The ability to tune the filter algorithm specifically to our test setup was critical to remove noise and correct estimates of IMUs, as others have reported [12].

AC and DMP also demonstrated sufficiently accurate and consistent but slightly poorer performance than the previously mentioned methods. AC displayed no drifting behavior and slightly higher errors due to the sole dependence on accelerometers that are sensitive to external accelerations (e.g., sudden acceleration/deceleration of the IMU) irrelevant to the gravity vector [42]. It is difficult to discuss the details of the DMP since the inner workings of DMP were proprietary. The default parameter values for DMP were not tunable and thus could not be optimized for our test setup. Others also have noted the importance of adjusting parameters for the given specific experimental scenarios [19], [20].

GI using high-pass-filtered gyroscopic signals displayed no drifting behavior and high enough accuracy and consistency for all speeds. The high-pass-filtered gyroscopic signals were useful for obtaining accurate and nondrifting angles since the high-pass filter inherently removes the gyroscopic bias [dc (=0 Hz) offset of a signal]. This bias instability of the gyroscopes is one of the main causes of the drifting behavior of gyroscopes [61]. Interestingly, a single high-pass filter could be used on different movement speeds ranging from 50°/s to 300°/s to provide drift-free and accurate GI angles.

B. Yaw

GI, BCF, KF, MW, and MH demonstrated stable and accurate computation of angles across all speeds during yaw rotations for the entire trial albeit in the absence of magnetometers (Figs. 4 and 5, bottom). For these methods, only high-pass-filtered gyroscopic signals were used to remove the gyroscopic bias since accelerometer readings could not be used to correct the gyroscopic drift. Thus, the use of a high-pass filter was necessary since the drift could not be compensated due to the absence of magnetometers, which provide the reference for the heading. The accelerometer could not read any meaningful signals related to the motion of the IMU but the gravity vector in the z-axis.

Filter parameters for BCF, KF, MW, and MH were modified to minimize the effect of accelerometers on the angle computation. For BCF, γ was set to close to zero such that the majority of computation was based on gyroscopic signals. For KF, $\mathbf{R}$ was set to a larger value to minimize the dependency on accelerometers. For MW and MH, β, K_p, and K_i were all set to zero to compute angles solely using the gyroscopic signals. Since MW and MH computed angles from gyroscopic signals using identical equations (i.e., integrating the quaternion derivative of the IMU and converting the quaternion back to Euler angles), the computed yaw angle from MW and MH was identical. Thus, the RMSE for MW and MH was identical for yaw angles.

AC and DMP could not be used for computing yaw angles. AC could not be implemented since the rotation axis was parallel to the gravity vector, making the accelerometer reading impossible to quantify the relevant motion of the IMU. DMP yaw angles drifted significantly from the beginning, exceeding the RMSE threshold after minutes of running time. The accuracy of DMP was worse than the other fusion filters since DMP uses a default set of parameters for its algorithm which may not guarantee accurate results in our experimental setup. The ability to tune filter parameters is essential for achieving accurate angles [12].

If the z-axis of the IMUs were not parallel to gravity, the computation of the yaw angle would be similar to roll and pitch angles: AC and DMP could be viable options, but the filter parameters of BCF, KF, MW, and MH do not need to be modified, and all methods except for GI would not require high-pass filtering of GI. Conversely, had the x- or y-axes been parallel to gravity then angle calculations for roll or pitch would become difficult and require modifications in preprocessing and filter parameters. Recall that roll, pitch, and yaw are just rotational terms relative to the body, not gravity.

C. Computational Execution Time

The computational execution time for nonsensor fusion algorithms (i.e., GI, AC) was significantly faster than the other sensor fusion algorithms (i.e., BCF, KF, MW, MH). The BCF, a relatively simple sensor fusion algorithm, required the computation of GI and AC simultaneously, increasing its computational time. The KF iteratively and optimally computed the Kalman gain at every time step, which involved more computationally heavy matrix operations, increasing the computational time even more. Finally, the advanced sensor fusion algorithms (i.e., MW, MH) required the most computational time due to the heavy computation of optimization using GD algorithms (MW), more estimations and corrections of the error terms and reference vectors (MH), and conversion from quaternion to Euler angles (MW, MH). Also, the MW and MH computed the full 3-D orientation of the IMUs, whereas KF only calculated a single angle at a time, which resulted in higher computational time for MW and MH. Thus, depending on the available computational resources, one may select more advanced (e.g., KF, MW, MH) or simpler algorithms (e.g., GI, AC, BCF).

D. Notes on High-Pass Filtering of Gyroscopic Signals

A few key points need to be considered when applying a high-pass filter to the raw gyroscopic signals. First, the use of a high-pass filter should be used as a last resort to remove the gyroscopic drift behavior, and the use of sensor fusion algorithms is preferred. This is because high-pass filtered gyroscopic signals cannot be used to estimate static angles, but only the change in angle between two instants of time, therefore not suitable for applications requiring the knowledge of the IMU static angle (e.g., gravity compensation of human limb in the field of biomechanics).

Second, a proper selection of high-pass filter type, order, and cut-off frequency is critical for removing the gyroscopic bias while maintaining the integrity of the signals of interest. In practice, the cut-off frequency of the high-pass filter for raw gyroscopic signals can be set below the lowest frequency of the motion experienced by the gyroscope. However, if the cut-off frequency is set too close to zero, edge effects (i.e., large spikes at the initial and end of the signal) and delays become larger [62]. If the cut-off frequency is too large, it may compromise the integrity of the signals of interest. Certain methods can be used to reduce the magnitude of edge effects [63], such as: 1) padding the data with the reflected version of the signal in the beginning and at the end of the signal or 2) including a small amount of "warm-up" time before conducting the experimental trial.

Once a high-pass filter is properly selected, it can be used to output accurate dynamic angle estimations for a range of speeds rather than a single speed. We demonstrated that, even though we utilized a single high-pass filter, the RMSE of yaw angles was below the acceptable threshold for all movement speeds for all methods except for AC and DMP.

E. Other Limitations and Future Work

Although this study was able to systematically validate various angle computational methods, there were some limitations.

First, our study only focused on computing the relative angle of a single rotation axis at a time. Interestingly, multiaxis rotation may only have a small effect on IMU accuracy as other studies reported similar accuracy during multiaxis versus single-axis rotations [9], [10]. Also, only one IMU was rotated, while the other was kept stationary. Thus, the accuracy of the computed angle depended on the static estimates from the stationary IMU and dynamic estimates of the moving IMU. This setup was appropriate in clinical applications (e.g., spasticity assessment about elbow joints), but may not be appropriate for applications involving extreme movements (e.g., jumping or sprinting). Second, the test apparatus did not have a moment arm to understand the effect of acceleration artifacts due to the offset between the rotation axis and IMU introduced to the system. While our previous studies displayed a similar level of IMU accuracy even when a moment arm was introduced [3], more rigorous testing may be needed to fully evaluate the performances of the computational methods. In cases where the IMUs experience significant motion-related disturbances that introduce a large amount of noise to IMUs, KF, MW, and MH may perform better than other methods since KF, MW, and MH not only fuse multiple sensor readings, but also removes the effect of these external accelerations by updating the gains of the filter every time step (KF) or using a GD-based algorithm (MW) or a PI compensator (MH). The AC and BCF do not have any algorithms for reducing the effect of large external accelerations like KF, MW, and MH, so the accuracy may suffer in dynamic conditions. Third, the maximum angular speed of the moving IMU was not high or low enough to simulate all possible movements observed in biomechanics. Higher speeds may result in higher error regardless of the rotation axis, as observed in other studies [9], [10]. This may be because a greater range of speed requires a different set of optimal filter parameters [12]. In addition, low movement speeds may require different filter parameters (e.g., lower cut-off frequency for a high-pass filter of gyroscopic readings) since the frequency of motion is lower. Fourth, this study did not test the IMU system on human subjects. In real-world biomechanics, the computation of joint angles becomes more complex: 1) IMU axes may not be parallel to the body segment of interest and move separately from the body segment due to soft tissue, irregular muscle geometry, and skin motion artifacts [1]; 2) the rotation axis of human joints is not always clear and may change during motion [12]; and 3) the IMU axes may not be always parallel to the underlying anatomical axes [64]. Fifth, the filter parameters in this study were specifically tuned for our test setup. These parameters should be custom-tuned for different applications, as others have reported [12]. For different applications and models of IMUs, movement speeds, and rotation axis, the filter parameters may be completely different. Lastly, the presented methods in this study required the two 6-axis IMUs to be parallel throughout the data acquisition, which may present a practical limitation due to the irregular shapes of the human body. Other calibration methods can be explored [1]. Albeit these limitations, the proposed 6-axis IMU system with these algorithms can be valuable for certain biomechanical and clinical applications (e.g., slow gait analysis, knee/elbow

flexion/extension assessments, spasticity assessment in which one body segment is moving while the other is relatively stationary.

For future work, other advanced estimation algorithms ([17], [18], [65], [66], [67]) that utilize kinematic constraints or spatial relationships between the IMUs can be investigated. These algorithms exploit the kinematic constraints (e.g., ball-socket or hinge joint) to accurately compute relative orientation between two 6-axis IMUs. In addition, the use of combinations of different estimation algorithms for the moving and stationary IMUs can be explored. For example, one may utilize AC (i.e., a simple, drift-free, computationally light, and accurate method) and use only accelerometer readings for computing the orientation of the stationary IMU while using more advanced sensor fusion algorithms that use both gyroscopic and accelerometer readings for computing the orientation of the moving IMU.

V. CONCLUSION

This study aimed to assist researchers new to the field of IMUs by providing a basic understanding of different IMU algorithms and an open-source code of the algorithms. In addition, the study systematically evaluated the performances of various angle computational methods (GI, AC, BCF, KF, DMP, MW, and MH) using 6-axis IMUs without magnetometers on a test apparatus following a strict testing protocol. Most (GI, BCF, KF, MW, and MH for yaw) if not all methods (for pitch/roll) accurately computed relative angles that were within the acceptable RMSE range of 6° at all speeds and test duration. For yaw, filter parameters were modified, and high-pass filtered gyroscopes were used. Even simple algorithms such as GI can output accurate and drift-free angles for various speed and rotation axes by properly preprocessing the raw signals. Proper filtering parameters for methods and preprocessing are essential and can be done by utilizing the information (e.g., movement speed, rotation axis) of the motion that IMU experiences.

REFERENCES

[1] T. Seel, J. Raisch, and T. Schauer, "IMU-based joint angle measurement for gait analysis," *Sensors*, vol. 14, no. 4, pp. 6891–6909, Apr. 2014, doi: 10.3390/s140406891.

[2] A. Brennan, J. Zhang, K. Deluzio, and Q. Li, "Quantification of inertial sensor-based 3D joint angle measurement accuracy using an instrumented gimbal," *Gait Posture*, vol. 34, no. 3, pp. 320–323, 2011, doi: 10.1016/j.gaitpost.2011.05.018.

[3] S. Y. Song, Y. Pei, S. R. Tippett, D. Lamichhane, C. M. Zallek, and E. T. Hsiao-Wecksler, "Validation of a wearable position, velocity, and resistance meter for assessing spasticity and rigidity," in *Proc. Frontiers Biomed. Devices*, 2018, pp. 1–4, Paper V001T10A007, doi: 10.1115/DMD2018-6906.

[4] P. Picerno, "25 years of lower limb joint kinematics by using inertial and magnetic sensors: A review of methodological approaches," *Gait Posture*, vol. 51, pp. 239–246, Jan. 2017, doi: 10.1016/j.gaitpost.2016.11.008.

[5] S. Y. Song, Y. Pei, J. Liang, and E. T. Hsiao-Wecksler, "Design of a portable position, velocity, and resistance meter (PVRM) for convenient clinical evaluation of spasticity or rigidity," in *Proc. Design Med. Devices Conf.*, Apr. 2017, pp. 1–2, Paper V001T11A020, doi: 10.1115/DMD2017-3503.

[6] T. Endo, N. Yoshikawa, H. Fujimura, and S. Sakoda, "Parkinsonian rigidity depends on the velocity of passive joint movement," *Parkinson's Disease*, vol. 2015, pp. 1–4, Dec. 2015, doi: 10.1155/2015/961790.

[7] R. Vitali *et al.*, "Method for estimating three-dimensional knee rotations using two inertial measurement units: Validation with a coordinate measurement machine," *Sensors*, vol. 17, no. 9, p. 1970, Aug. 2017, doi: 10.3390/s17091970.

[8] J. Favre, B. M. Jolles, R. Aissaoui, and K. Aminian, "Ambulatory measurement of 3D knee joint angle," *J. Biomech.*, vol. 41, no. 5, pp. 1029–1035, Jan. 2008, doi: 10.1016/j.jbiomech.2007.12.003.

[9] K. Lebel, P. Boissy, M. Hamel, and C. Duval, "Inertial measures of motion for clinical biomechanics: Comparative assessment of accuracy under controlled conditions—Effect of velocity," *PLoS ONE*, vol. 8, no. 11, Nov. 2013, Art. no. e79945, doi: 10.1371/journal.pone.0079945.

[10] L. Ricci, F. Taffoni, and D. Formica, "On the orientation error of IMU: Investigating static and dynamic accuracy targeting human motion," *PLoS ONE*, vol. 11, no. 9, Sep. 2016, Art. no. e0161940, doi: 10.1371/journal.pone.0161940.

[11] E. Bergamini, P. Guillon, V. Camomilla, H. Pillet, W. Skalli, and A. Cappozzo, "Trunk inclination estimate during the sprint start using an inertial measurement unit: A validation study," *J. Appl. Biomech.*, vol. 29, no. 5, pp. 622–627, Oct. 2013, doi: 10.1123/jab.29.5.622.

[12] M. Caruso, A. M. Sabatini, M. Knaflitz, M. Gazzoni, U. D. Croce, and A. Cereatti, "Orientation estimation through magneto-inertial sensor fusion: A heuristic approach for suboptimal parameters tuning," *IEEE Sensors J.*, vol. 21, no. 3, pp. 3408–3419, Feb. 2021, doi: 10.1109/JSEN.2020.3024806.

[13] M. Caruso, A. M. Sabatini, M. Knaflitz, U. D. Croce, and A. Cereatti, "Extension of the rigid-constraint method for the heuristic suboptimal parameter tuning to ten sensor fusion algorithms using inertial and magnetic sensing," *Sensors*, vol. 21, no. 18, p. 6307, Sep. 2021, doi: 10.3390/s21186307.

[14] W. H. K. de Vries, H. E. J. Veeger, C. T. M. Baten, and F. C. T. van der Helm, "Magnetic distortion in motion labs, implications for validating inertial magnetic sensors," *Gait Posture*, vol. 29, no. 4, pp. 535–541, 2009, doi: 10.1016/j.gaitpost.2008.12.004.

[15] E. R. Bachmann, X. Yun, and A. Brumfield, "Investigating the effects of magnetic variations on inertial/magnetic orientation sensors," Nav. Postgraduate School, Monterey, CA, USA, Tech. Rep. ADA601163, 2007.

[16] G. Ligorio and A. M. Sabatini, "A novel Kalman filter for human motion tracking with an inertial-based dynamic inclinometer," *IEEE Trans. Biomed. Eng.*, vol. 62, no. 8, pp. 2033–2043, Aug. 2015, doi: 10.1109/TBME.2015.2411431.

[17] I. Weygers *et al.*, "Drift-free inertial sensor-based joint kinematics for long-term arbitrary movements," *IEEE Sensors J.*, vol. 20, no. 14, pp. 7969–7979, Mar. 2020.

[18] D. Lehmann, D. Laidig, R. Deimel, and T. Seel, "Magnetometer-free inertial motion tracking of arbitrary joints with range of motion constraints," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 16016–16022, Jan. 2020.

[19] M. Caruso *et al.*, "Analysis of the accuracy of ten algorithms for orientation estimation using inertial and magnetic sensing under optimal conditions: One size does not fit all," *Sensors*, vol. 21, no. 7, p. 2543, Apr. 2021, doi: 10.3390/s21072543.

[20] D. Laidig, M. Caruso, A. Cereatti, and T. Seel, "BROAD—A benchmark for robust inertial orientation estimation," *Data*, vol. 6, no. 7, p. 72, Jun. 2021, doi: 10.3390/data6070072.

[21] P. Müller, M.-A. S. T. Bégin, and T. Seel, "Alignment-free, self-calibrating elbow angles measurement using inertial sensors," *IEEE J. Biomed. Health Inform.*, vol. 21, no. 2, pp. 312–319, Dec. 2017.

[22] A. Leardini, G. Lullini, S. Giannini, L. Berti, M. Ortolani, and P. Caravaggi, "Validation of the angular measurements of a new inertial-measurement-unit based rehabilitation system: Comparison with state-of-the-art gait analysis," *J. Neuroeng. Rehabil.*, vol. 11, no. 1, p. 136, 2014, doi: 10.1186/1743-0003-11-136.

[23] N. Ghotbi, N. N. Ansari, S. Naghdi, S. Hasson, B. Jamshidpour, and S. Amiri, "Inter-rater reliability of the modified modified Ashworth scale in assessing lower limb muscle spasticity," *Brain Injury*, vol. 23, no. 10, pp. 815–819, Jan. 2009, doi: 10.1080/02699050903200548.

[24] A. H. Bakhtiary and E. Fatemy, "Does electrical stimulation reduce spasticity after stroke? A randomized controlled study," *Clin. Rehabil.*, vol. 22, no. 5, pp. 418–425, May 2008, doi: 10.1177/0269215507084008.

[25] C.-L. Wu, M.-H. Huang, C.-L. Lee, C.-W. Liu, L.-J. Lin, and C.-H. Chen, "Effect on spasticity after performance of dynamic-repeated-passive ankle joint motion exercise in chronic stroke patients," *Kaohsiung J. Med. Sci.*, vol. 22, no. 12, pp. 610–617, Dec. 2006, doi: 10.1016/S1607-551X(09)70361-4.

[26] H.-M. Lee, J.-J.-J. Chen, M.-S. Ju, C.-C.-K. Lin, and P. P. W. Poon, "Validation of portable muscle tone measurement device for quantifying velocity-dependent properties in elbow spasticity," *J. Electromyogr. Kinesiol.*, vol. 14, no. 5, pp. 577–589, Oct. 2004, doi: 10.1016/J.JELEKIN.2004.02.002.

[27] A. F. Thilmann, S. J. Fellows, and E. Garms, "The mechanism of spastic muscle hypertonus. Variation in reflex gain over the time course of spasticity," *Brain*, vol. 114, no. 1, pp. 233–244, Feb. 1991.

[28] S. Malhotra *et al.*, "An investigation into the agreement between clinical, biomechanical and neurophysiological measures of spasticity," *Clin. Rehabil.*, vol. 22, no. 12, pp. 1105–1115, Dec. 2008, doi: 10.1177/0269215508095089.

[29] D. A. Winter, "Biomechanical motor patterns in normal walking," *J. Motor Behav.*, vol. 15, no. 4, pp. 302–330, Dec. 1983, doi: 10.1080/00222895.1983.10735302.

[30] B. F. Mentiplay, M. Banky, R. A. Clark, M. B. Kahn, and G. Williams, "Lower limb angular velocity during walking at various speeds," *Gait Posture*, vol. 65, pp. 190–196, Sep. 2018, doi: 10.1016/j.gaitpost.2018.06.162.

[31] B. R. Umberger and P. E. Martin, "Mechanical power and efficiency of level walking with different stride rates," *J. Exp. Biol.*, vol. 210, no. 18, pp. 3255–3265, Sep. 2007, doi: 10.1242/jeb.000950.

[32] S. Stančin and S. Tomažič, "Time- and computation-efficient calibration of MEMS 3D accelerometers and gyroscopes," *Sensors*, vol. 14, no. 8, pp. 14885–14915, Aug. 2014, doi: 10.3390/s140814885.

[33] E. H. Shin and N. El-Sheimy, "A new calibration method for strapdown inertial navigation systems," *Z. Vermess*, vol. 127, pp. 1–10, Jan. 2002.

[34] J. Diebel, "Representing attitude: Euler angles, unit quaternions, and rotation vectors," *Matrix*, vol. 58, nos. 15–16, pp. 1–35, 2006.

[35] G. B. Arfken and H. J. Weber, "Mathematical methods for physicists," *Amer. J. Phys.*, vol. 67, no. 2, p. 165, Jan. 1999.

[36] MathWorks. (Jan. 8, 2021). *Quat2Eul*. [Online]. Available: https://www.mathworks.com/help/robotics/ref/quat2eul.html

[37] *MPU-6050 Six-Axis (Gyro + Accelerometer) MEMS MotionTracking*™ *Devices*. Accessed: Apr. 21, 2019. [Online]. Available: https://www.invensense.com/products/motion-tracking/6-axis/mpu-6050/

[38] M. El-Gohary and J. McNames, "Human joint angle estimation with inertial sensors and validation with a robot arm," *IEEE Trans. Biomed. Eng.*, vol. 62, no. 7, pp. 1759–1767, Jul. 2015, doi: 10.1109/TBME.2015.2403368.

[39] J. Du, C. Gerdtman, and M. Lindén, "Signal processing algorithms for temperauture drift in a MEMS-gyro-based head mouse," in *Proc. IWSSIP*, 2014, pp. 123–126.

[40] J. Du, C. Gerdtman, and M. Lindén, "Signal quality improvement algorithms for MEMS gyroscope-based human motion analysis systems: A systematic review," *Sensors*, vol. 18, no. 4, p. 1123, Apr. 2018, doi: 10.3390/s18041123.

[41] J. Favre, F. Luthi, B. M. Jolles, O. Siegrist, B. Najafi, and K. Aminian, "A new ambulatory system for comparative evaluation of the three-dimensional knee kinematics, applied to anterior cruciate ligament injuries," *Knee Surg., Sports Traumatol., Arthroscopy*, vol. 14, no. 7, pp. 592–604, Jul. 2006.

[42] P. Gui, L. Tang, and S. Mukhopadhyay, "MEMS based IMU for tilting measurement: Comparison of complementary and Kalman filter based data fusion," in *Proc. IEEE 10th Conf. Ind. Electron. Appl. (ICIEA)*, Jun. 2015, pp. 2004–2009, doi: 10.1109/ICIEA.2015.7334442.

[43] N. Ahmad, R. A. R. Ghazilla, N. M. Khairi, and V. Kasi, "Reviews on various inertial measurement unit (IMU) sensor applications," *Int. J. Signal Process. Syst.*, vol. 1, no. 2, pp. 256–262, 2013.

[44] F. Alam, Z. ZhaiHe, and H. JiaJia, "A comparative analysis of orientation estimation filters using MEMS based IMU," in *Proc. Int. Conf. Res. Sci., Eng. Technol.*, Dubai, UAE, 2014, pp. 21–22.

[45] C. J. Fisher, "Using an accelerometer for inclination sensing," Analog Devices, Cambridge, MA, USA, Appl. Note AN-1057, Feb. 2010, pp. 1–8.

[46] H. J. Luinge and P. H. Veltink, "Inclination measurement of human movement using a 3-D accelerometer with autocalibration," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 12, no. 1, pp. 112–121, Mar. 2004, doi: 10.1109/TNSRE.2003.822759.

[47] S. O. H. Madgwick, A. J. L. Harrison, and R. Vaidyanathan, "Estimation of IMU and MARG orientation using a gradient descent algorithm," in *Proc. IEEE Int. Conf. Rehabil. Robot.*, May 2011, pp. 1–7.

[48] W. Higgins, "A comparison of complementary and Kalman filtering," *IEEE Trans. Aerosp. Electron. Syst.*, vol. AES-11, no. 3, pp. 321–325, May 1975, doi: 10.1109/TAES.1975.308081.

[49] R. Valenti, I. Dryanovski, and J. Xiao, "Keeping a good attitude: A quaternion-based orientation filter for IMUs and MARGs," *Sensors*, vol. 15, no. 8, pp. 19302–19330, Aug. 2015, doi: 10.3390/s150819302.

[50] S. Colton, "The balance filter," Massachusetts Inst. Technol., Cambridge, MA, USA, Tech. Rep., 2007. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/7334442

[51] K. J. O'Donovan, R. Kamnik, D. T. O'Keeffe, and G. M. Lyons, "An inertial and magnetic sensor based technique for joint angle measurement," *J. Biomech.*, vol. 40, no. 12, pp. 2604–2611, 2007, doi: 10.1016/j.jbiomech.2006.12.010.

[52] R. E. Kalman, "A new approach to linear filtering and prediction problems," *J. Basic Eng.*, vol. 82, no. 1, pp. 35–45, Mar. 1960.

[53] Y. Kim and H. Bang, "Introduction to Kalman filter and its applications," in *Introduction and Implementations of the Kalman Filter*, F. Govaers, Ed. London, U.K.: IntechOpen, 2018.

[54] VectorNav. *IMU Specifications*. Accessed: Dec. 5, 2021. [Online]. Available: https://www.vectornav.com/resources/inertial-navigation-primer/specifications–and–error-budgets/specs-imuspecs

[55] S. A. Ludwig and K. D. Burnham, "Comparison of Euler estimate using extended Kalman filter, Madgwick and Mahony on quadcopter flight data," in *Proc. Int. Conf. Unmanned Aircr. Syst. (ICUAS)*, Jun. 2018, pp. 1236–1241, doi: 10.1109/ICUAS.2018.8453465.

[56] S. O. H. Madgwick, S. Wilson, R. Turk, J. Burridge, C. Kapatos, and R. Vaidyanathan, "An extended complementary filter for full-body MARG orientation estimation," *IEEE/ASME Trans. Mechatronics*, vol. 25, no. 4, pp. 2054–2064, Aug. 2020, doi: 10.1109/TMECH.2020.2992296.

[57] R. Mahony, T. Hamel, and J.-M. Pflimlin, "Nonlinear complementary filters on the special orthogonal group," *IEEE Trans. Autom. Control*, vol. 53, no. 5, pp. 1203–1218, Jun. 2008, doi: 10.1109/TAC.2008.923738.

[58] T. Michel, H. Fourati, P. Geneves, and N. Layaida, "A comparative analysis of attitude estimation for pedestrian navigation with smartphones," in *Proc. Int. Conf. Indoor Positioning Indoor Navigat. (IPIN)*, Oct. 2015, pp. 1–10.

[59] R. S. McGinnis, S. M. Cain, S. P. Davidson, R. V. Vitali, S. G. McLean, and N. C. Perkins, "Validation of complementary filter based IMU data fusion for tracking torso angle and rifle orientation," in *Proc. IMECE*, Nov. 2014, pp. 1–10, Paper V003T03A052, doi: 10.1115/imece2014-36909.

[60] T. Beravs, P. Rebersek, D. Novak, J. Podobnik, and M. Munih, "Development and validation of a wearable inertial measurement system for use with lower limb exoskeletons," in *Proc. 11th IEEE-RAS Int. Conf. Humanoid Robots*, Oct. 2011, pp. 212–217, doi: 10.1109/Humanoids.2011.6100914.

[61] V. M. N. Passaro, A. Cuccovillo, L. Vaiani, M. De Carlo, and C. E. Campanella, "Gyroscope technology and applications: A review in the industrial perspective," *Sensors*, vol. 17, no. 10, p. 2284, Oct. 2017, doi: 10.3390/s17102284.

[62] L. Tan and J. Jiang, *Subband-and Wavelet-Based Coding*, E. Jiang, Ed. Boston, MA, USA: Academic, 2013, ch. 13, pp. 621–681.

[63] M. X. Cohen and X. Michael, *Analyzing Neural Time Series Data: Theory and Practice*. Cambridge, MA, USA: MIT, 2014. [Online]. Available: http://search.ebscohost.com/login.aspx?direct=true&scope=site&db=nlebk&db=nlabk&AN=689432

[64] R. V. Vitali and N. C. Perkins, "Determining anatomical frames via inertial motion capture: A survey of methods," *J. Biomech.*, vol. 106, Jun. 2020, Art. no. 109832, doi: 10.1016/j.jbiomech.2020.109832.

[65] J. Lee and M. Choi, "Robust inertial measurement unit-based attitude determination Kalman filter for kinematically constrained links," *Sensors*, vol. 19, no. 4, p. 768, Feb. 2019.

[66] K. Eckhoff, M. Kok, S. Lucia, and T. Seel, "Sparse magnetometer-free inertial motion tracking—A condition for observability in double Hinge joint systems," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 16023–16030, 2020.

[67] A. Zedda *et al.*, "DoMoMEA: A home-based telerehabilitation system for stroke patients," in *Proc. 42nd Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC)*, Jul. 2020, pp. 5773–5776, doi: 10.1109/EMBC44109.2020.9175742.
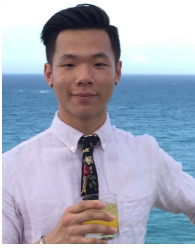
Seung Yun Song was born in Busan, South Korea, in 1992. He received the B.S. degree in materials science and engineering from Purdue University, West Lafayette, IN, USA, in 2015, and the M.S. degree in mechanical engineering from the University of Illinois at Urbana-Champaign, Champaign, IL, USA, in 2019, where he is currently pursuing the Ph.D. degree in mechanical engineering with the Department of Mechanical Science and Engineering.

He worked as a Mechanical Engineer with LifeFoundry, Inc., Champaign. His main research areas include wearable technology, medical devices, human–robot interfaces, and robotics.

Yinan Pei (Student Member, IEEE) was born in Beijing, China, in 1992. He received the B.S. degree in civil engineering from the University of California at Los Angeles, Los Angeles, CA, USA, in 2015, and the M.S. degree in mechanical engineering from the University of Illinois at Urbana-Champaign, Champaign, IL, USA, in 2018, where he is currently pursuing the Ph.D. degree in mechanical engineering with the Department of Mechanical Science and Engineering.

His main research areas are medical robotics, wearable sensors, and biomechanics.

Elizabeth T. Hsiao-Wecksler (Member, IEEE) received the B.S. degree from Cornell University, Ithaca, NY, USA, in 1987, the M.S. degree from the Rochester Institute of Technology, Henrietta, NY, in 1994, and the Ph.D. degree from the University of California at Berkeley, Berkeley, CA, USA, in 2000, all in mechanical engineering.

She held a Postdoctoral Fellowship in the Integrated Rehabilitation Engineering Program at Boston University, Boston, MA, USA, and Harvard Medical School, Boston. She is a Professor with the Department of Mechanical Science and Engineering, University of Illinois at Urbana-Champaign, Urbana, IL, USA, where she also directs the Human Dynamics and Controls Laboratory. Her research interests include the design and control of assistive mobility devices and medical education task trainers, wearable technologies, and locomotion biomechanics.

Dr. Hsiao-Wecksler is a Fellow of ASME and an Associate Editor of the *Journal of Medical Devices* (ASME). She is also a Fellow and the Past President of the American Society of Biomechanics.