IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 29 September 2022, accepted 23 October 2022, date of publication 26 October 2022, date of current version 4 November 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3217555

RESEARCH ARTICLE

Orientation Prediction for VR and AR Devices Using Inertial Sensors Based on Kalman-Like Error Compensation

LE THI HUE DAO[1], (Student Member, IEEE), TRUONG THANH NHAT MAI[1], (Student Member, IEEE), WOOK HONG[2], SANGHYUN PARK[2], HOKWON KIM[2], JOON GOO LEE[2], MIN-SEOK KIM[2], AND CHUL LEE[1], (Member, IEEE)

[1]Department of Multimedia Engineering, Dongguk University, Seoul 04620, South Korea
[2]Research and Development Group, RAONTECH, Seongnam-si 13567, South Korea

Corresponding author: Chul Lee (chullee@dongguk.edu)

This work was supported in part by the Institute of Information and Communications Technology Planning Evaluation (IITP) Grant by the Korea Government (MSIT) (Development of Low MTP Latency Microdisplay and SoC Technology for VR·AR Devices) under Grant 2020-0-00852; and in part by the Culture, Sports and Tourism Research and Development Program through the Korea Creative Content Agency (KOCCA) Grant by the Korea Government (MCST) under Grant R2022020120.

ABSTRACT We propose an orientation prediction algorithm based on Kalman-like error compensation for virtual reality (VR) and augmented reality (AR) devices using measurements of an inertial measurement unit (IMU), which includes a tri-axial gyroscope and a tri-axial accelerometer. First, the initial prediction of the orientation is estimated by assuming linear movement. Then, to improve the prediction accuracy, the accuracies of previous predictions are taken into account by computing the orientation difference between the current orientation and previous prediction. Finally, we define a weight matrix to determine the optimal adjustments for predictions corresponding to a given orientation, which is obtained by minimizing the estimation errors based on the minimum mean square error (MMSE) criterion using Kalman-like error compensation. Experimental results demonstrate that the proposed algorithm exhibits higher orientation prediction accuracy compared with conventional algorithms on several open datasets.

INDEX TERMS Orientation prediction, inertial measurement units (IMUs), motion-to-photon (MTP) latency, virtual reality (VR), augmented reality (AR), attitude and heading reference system (AHRS), minimum mean square error (MMSE).

I. INTRODUCTION

Augmented reality (AR) and virtual reality (VR) technologies, which either enrich or replace real-world environments with simulated ones, have recently garnered significant attention in both industry and academia [1], [2], [3], [4], [5], [6]. However, VR and AR systems suffer from end-to-end delay or latency, which affects user experience. In particular, motion-to-photon (MTP) latency, which is defined as the time between a user's movement and the corresponding movement rendered on the screen of a head-mounted display, is a major challenge that limits the applicability of VR and AR [7]. For example, MTP latencies higher than 20 ms induce motion sickness or dizziness in users [8], [9]. Therefore, MTP latency reduction is essential for improving users' virtual experience. A common approach to reduce MTP latency is to predict future head orientations by anticipating a user's movements, which enables VR and AR devices to render future scenes in advance. Therefore, it is essential to develop algorithms capable of accurately predicting head orientation in VR and AR systems. In addition to AR and VR, orientation prediction plays a vital role in various practical applications, such as unmanned aerial vehicles [10], robotics [11], [12], and navigation systems [13], [14], [15].

The associate editor coordinating the review of this manuscript and approving it for publication was Yiqi Liu.

Orientation prediction requires the prior estimation of current orientation, which can be estimated using information from various measurement sources, *e.g.*, inertial sensors or mono/stereo cameras. For example, in [13], [16], [17], [18], [19], and [20], data measured from an inertial sensor and a camera were merged to estimate the orientation of an object, where an inertial sensor was used to determine orientation by integrating measurements, while camera outputs were used to compensate for errors. However, the use of visual information increases the complexity of the process and makes it unsuitable for resource-constrained devices. Further, changes in illumination affect the performance of camera-based algorithms.

The advancement of MEMS technology have led to the development of small and inexpensive inertial sensors called inertial measurement units (IMUs). Because of their high sampling rates, minimal latency, and small size [21], [22], IMUs have been employed in modern VR and AR devices to acquire data for localization of locations and orientations. A commercial IMU comprises a tri-axial gyroscope, a tri-axial accelerometer, and/or a magnetometer. The gyroscope measures angular velocity and integrates it over time to determine orientation angles. However, the integration process also accumulates sensor errors; thus, the estimated orientations deviate from the true orientation with time, making them unreliable [23]. The accelerometer is used to overcome this issue by measuring the Earth's gravitational field, thereby yielding a reference vector for correcting the estimated orientations. Magnetometers, which measure the Earth's magnetic field, are also used for correcting angular velocity. However, because of their high sensitivity to magnetic disturbance, such as those arising from metal or electrical equipment, their applicability is limited [24].

Several orientation prediction algorithms for VR and AR devices have been developed using IMUs based on Kalman filters (KFs) and particle filters (PFs) [25], [26], [27], [28], [29]. KF-based algorithms derive optimal estimators by recursively performing two steps—prediction and correction. In the prediction step, the filter predicts the future orientation along with its uncertainty. Once the actual measurement is observed, the prediction is corrected using weighted averaging, where estimates with lower uncertainty are assigned greater weights [30]. By contrast, a PF is a recursive Bayesian state estimator that represents the probability density function of a system state using random samples. Because PFs can utilize a state space model in any form, they can be applied to a wide range of models. However, despite their higher complexity, PFs do not exhibit significant improvement over KFs [31], [32]. Thus, KFs are more commonly employed in orientation estimation and prediction.

As the standard KF is an optimal estimator only under the assumption of linearity, its variants, *e.g.*, extended Kalman filter (EKF) and unscented Kalman filter (UKF), are widely used for orientation prediction in nonlinear systems [33]. In particular, EKF predicts orientation by linearizing state transition functions, which are calculated based on the previous estimate, to the current estimate. For instance, Zhao and Wang [34] compensated for deviations in estimates by fusing redundant data from ultrasonic sensors and magnetometers. Tong et al. [35] compensated for disturbances in the measurements to improve orientation estimation accuracy. Ghobadi et al. [36] used the covariance inflation technique to adjust noise covariance and reduce measurement uncertainty. Further, Park et al. [37] mitigated the effect of acceleration and magnetic disturbance by adjusting the measurement covariance using the ellipsoidal method. In [38], Sun et al, decoupled roll and pitch estimated from the magnetometer measurements to reduce the influence of the magnetic disturbances. However, when the state transition and observation models are highly nonlinear, EKF may exhibit inferior performance because of the propagation of uncertainty.

In UKF, linearization is replaced with a deterministic sampling scheme known as unscented transformation, which chooses a set of sample points around the mean. Several recent orientation estimation algorithms have employed UKF to improve estimation accuracy. For example, in [39], Marina et al. used a tri-axial attitude determination algorithm as the observation model. Chiella et al. [40] developed an adaptive strategy based on covariance matching to tune measurement covariance. Further, Tong et al. [41] divided the measurement updates of the gravity vector and magnetic field vector into two stages to avoid the undesirable correction of the Euler angle error. Although having a large number of sample points improves accuracy, it also increases the complexity of the algorithms; high computational loads are one of the most crucial drawbacks. Therefore, UKF-based algorithms are unsuitable for devices with limited computational power, such as VR and AR devices.

In this work, we propose a novel orientation prediction algorithm to improve prediction accuracy by addressing the limitations of conventional prediction algorithms. The proposed algorithm operates by estimating linear displacement from the current orientation and subsequent adjustments to compensate for the error caused by linear prediction. Current predictions are adjusted based on the accuracies of previous predictions. To this end, the orientation difference between the previous linear prediction and the current orientation is ascertained. Then, we define a weight matrix to determine the optimal adjustment of the current prediction based on the orientation difference. Similar to the derivation of KF—differences with lower uncertainty are assigned greater weights—the optimal weight matrix is obtained as the minimum mean square error (MMSE) estimator using the prediction error between the current and previously predicted orientations. Experimental results demonstrate that the proposed algorithm outperforms conventional orientation prediction algorithms on several open datasets.

In summary, the main contributions of this paper are as follows:

- We propose an orientation prediction algorithm that uses the accuracy of previous predictions to improve the current prediction accuracy.

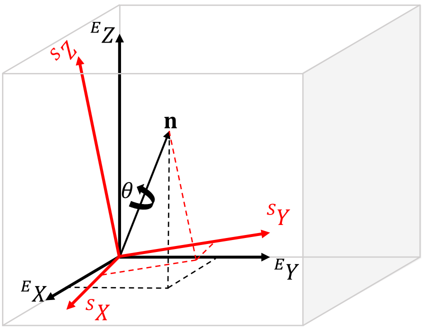


FIGURE 1. **Illustration of the quaternion as rotation of the earth frame *E* to the sensor frame *S* by angle θ around a vector n.**

- We develop a Kalman-like error compensation scheme to determine a weight matrix for optimal adjustment with respect to linear movement by minimizing the estimation error in the MMSE sense.
- The proposed algorithm is experimentally verified to outperform conventional orientation prediction algorithms on several open datasets.

The rest of this paper is organized as follows. Section II introduces the notations used in this paper. Section III describes the proposed orientation prediction algorithm, and Section IV discusses the experimental results. Finally, Section V concludes the paper.

II. BACKGROUND

In this section, we briefly introduce the notations used throughout this paper. Hereafter, scalar values are denoted by italicized letters, and matrices and vectors by bold roman uppercase and lowercase letters, respectively.

Two coordinate frames—the Earth frame E and the sensor frame S—are used to represent orientation in three-dimensional (3D) space. A rotation between the frames can be represented using a rotation matrix, Euler angles, or a unit quaternion. The rotation matrix is a 3×3 matrix that represents the orientation. In comparison, the unit quaternion represents orientation using only four elements. Therefore, computations using the rotation matrix are more time-consuming than those using the unit quaternion. Euler angles represent orientation using three angles—roll, pitch, and yaw—and they are commonly used because of their intuitive physical representation. However, they exhibit a phenomenon known as "gimbal lock," which causes a singularity problem in orientation representation [42]. Among the aforementioned representations, unit quaternions are more compact, numerically stable, and computationally efficient than rotation matrices. Moreover, the use of quaternions prevents the loss of degrees of freedom that appears when Euler angles are used [43]. Thus, unit quaternions are used to represent orientation in this paper.

The quaternion $\mathbf{q}$ can be used to represent the orientation of the sensor frame S relative to the Earth frame E. Specifically, the quaternion can be interpreted as a rotation around an arbitrary axis in 3D space, as depicted in Figure 1, and can also be considered to be a vector in a four-dimensional vector space [42], given by

$$\mathbf{q} = \left[\cos\frac{\theta}{2}, \mathbf{n}\sin\frac{\theta}{2}\right]^T = [q_0, q_1, q_2, q_3]^T, \tag{1}$$

where q_0 and $[q_1, q_2, q_3]^T$ are the scalar and vector parts of the quaternion, respectively, $\mathbf{n}$ denotes the normalized rotational axis, and θ is the angle of rotation.

The product of two quaternions, $\mathbf{q}$ and $\mathbf{p}$, is defined as [42]

$$\begin{aligned}\mathbf{q} \otimes \mathbf{p} &= [q_0, q_1, q_2, q_3]^T \otimes [p_0, p_1, p_2, p_3]^T \\ &= \begin{bmatrix} q_0p_0 - q_1p_1 - q_2p_2 - q_3p_3 \\ q_0p_1 + q_1p_0 + q_2p_3 - q_3p_2 \\ q_0p_2 - q_1p_3 + q_2p_0 + q_3p_1 \\ q_0p_3 + q_1p_2 - q_2p_1 + q_3p_0 \end{bmatrix}.\end{aligned} \tag{2}$$

Please note that the quaternion product is not commutative, *i.e.*, $\mathbf{q} \otimes \mathbf{p} \neq \mathbf{p} \otimes \mathbf{q}$.

Let us assume that the quaternion $\mathbf{q}$ represents the orientation of a rotating sensor frame relative to the reference Earth frame. Then, the quaternion rate or the time-derivative of this quaternion can be expressed in terms of the corresponding angular velocity vector $\boldsymbol{\omega}$, which defines the rate at which the sensor frame rotates relative to the Earth frame or the rate of change in the orientation. The quaternion derivative is defined as [44]

$$\dot{\mathbf{q}} = \frac{1}{2}\mathbf{q} \otimes \begin{bmatrix} 0 \\ \boldsymbol{\omega} \end{bmatrix} = \frac{1}{2}\Omega(\boldsymbol{\omega})\mathbf{q}, \tag{3}$$

where $\boldsymbol{\omega} = [\omega_x, \omega_y, \omega_z]^T$ represents the angular velocities of the x-, y-, and z-axes in the sensor frame; and $\Omega(\boldsymbol{\omega})$ denotes the right quaternion multiplier related to $\boldsymbol{\omega}$, defined as

$$\Omega(\boldsymbol{\omega}) = \begin{bmatrix} 0 & -\omega_x & -\omega_y & -\omega_z \\ \omega_x & 0 & \omega_z & -\omega_y \\ \omega_y & -\omega_z & 0 & \omega_x \\ \omega_z & \omega_y & -\omega_x & 0 \end{bmatrix}. \tag{4}$$

A vector $^{E}\mathbf{v} \in \mathbb{R}^3$ in the Earth frame can be rotated to the vector $^{S}\mathbf{v} \in \mathbb{R}^3$ in the sensor frame using the quaternion $\mathbf{q}$ as [44]

$$^{S}\mathbf{v} = \mathbf{R}(\mathbf{q})^{E}\mathbf{v}, \tag{5}$$

where the rotation matrix $\mathbf{R}(\mathbf{q})$ is defined as

$$\begin{aligned}&\mathbf{R}(\mathbf{q}) \\ &= \begin{bmatrix} q_0^2+q_1^2-q_2^2-q_3^2 & 2(q_1q_2+q_0q_3) & 2(q_1q_3-q_0q_2) \\ 2(q_1q_2-q_0q_3) & q_0^2-q_1^2+q_2^2-q_3^2 & 2(q_2q_3+q_0q_1) \\ 2(q_1q_3+q_0q_2) & 2(q_2q_3-q_0q_1) & q_0^2-q_1^2-q_2^2+q_3^2 \end{bmatrix}.\end{aligned} \tag{6}$$

Quaternion arithmetic requires quaternion describing orientation to be of unit length. Therefore, quaternions are first normalized as

$$|\mathbf{q}| = \sqrt{q_0^2 + q_1^2 + q_2^2 + q_3^2} = 1. \tag{7}$$

In this paper, all quaternions used for representing orientation are taken to be unit quaternions, *i.e.*, $\mathbf{q}^T\mathbf{q} = 1$.

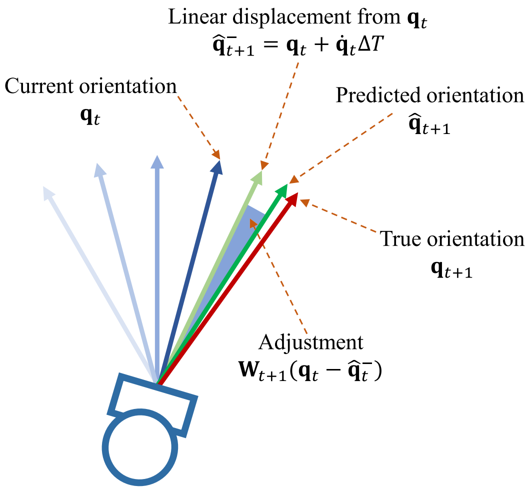

Linear displacement from $\mathbf{q}_t$
$\widehat{\mathbf{q}}^-_{t+1} = \mathbf{q}_t + \dot{\mathbf{q}}_t \Delta T$
Current orientation
$\mathbf{q}_t$
Predicted orientation
$\widehat{\mathbf{q}}_{t+1}$
True orientation
$\mathbf{q}_{t+1}$
Adjustment
$\mathbf{W}_{t+1}(\mathbf{q}_t - \widehat{\mathbf{q}}^-_t)$

FIGURE 2. **Illustration of prediction of head orientation.**

III. PROPOSED ORIENTATION PREDICTION ALGORITHM

In this section, an overview of the proposed algorithm is first provided. Then, the details of the proposed algorithm to predict future orientations are described.

A. OVERVIEW

Traditional EKF- and UKF-based algorithms [45], [46] can correct orientation prediction using only accelerometer readings. However, a greater amount of yaw information of orientations can be obtained from magnetometer readings. Therefore, many extensions of EKF- and UKF-based algorithms [34], [35], [36], [37], [38], [39], [40], [41] that use additional information from magnetometer readings to complement the accelerometer readings during correction have been developed. However, these algorithms introduce additional computational complexity as they require the processing of more information. In this work, to avoid additional computations, we develop an efficient yet effective orientation prediction algorithm, which uses only gyroscope and accelerometer readings.

Figure 2 shows an overview of the proposed algorithm, which comprises the estimation of linear displacement from the current orientation and subsequent adjustment. In Figure 2, head orientations at two consecutive time steps t and $(t + 1)$ are visualized. At time t, the current orientation $\mathbf{q}_t$ is obtained as indicated by the dark blue arrow. The linear displacement $\widehat{\mathbf{q}}^-_{t+1}$ is estimated, represented by the light green arrow, based on the assumption that the head rotates linearly with a constant angular velocity. However, movement can also be nonlinear. Therefore, an adjustment $\mathbf{W}_{t+1}(\mathbf{q}_t - \widehat{\mathbf{q}}^-_t)$ is required to compensate for the error caused by linear prediction by considering the accuracy of the previous linear prediction. Finally, the future orientation $\widehat{\mathbf{q}}_{t+1}$, denoted by the dark green arrow, is determined by combining the linear displacement $\widehat{\mathbf{q}}^-_{t+1}$ and the adjustment $\mathbf{W}_{t+1}(\mathbf{q}_t - \widehat{\mathbf{q}}^-_t)$.

Figure 3 shows the functional diagram of the proposed algorithm to estimate $\widehat{\mathbf{q}}_{t+1}$, which consists of six components—tracking, compensation, linear motion estimation, weight matrix computation, adjustment computation, and prediction. Note that the true head orientation at time t is not given, but it is estimated using a tracking algorithm. Therefore, tracking estimates the current head orientation $\overline{\mathbf{q}}_t$ with respect to the global reference frame. As the main goal of this work is the prediction of the future orientation $\widehat{\mathbf{q}}_{t+1}$, any conventional tracking algorithm can be used as a tracker. In this work, we employ the filter developed in [47] because of its high accuracy and efficiency. The compensation process corrects the gyroscope output using the tracked orientation $\overline{\mathbf{q}}_t$ to obtain $\overline{\boldsymbol{\omega}}_t$. Both these components use the accelerometer and gyroscope outputs, $\mathbf{a}_t$ and $\boldsymbol{\omega}_t$, respectively, as input. Following tracking and compensation, the linear motion estimation component uses the output of the two components to estimate the linear displacement. Next, the weight matrix computation component computes the optimal weight matrix for adjustment. Then, the adjustment computation component uses the computed weight matrix and the orientation difference between the current tracked orientation and the preceding linear prediction to generate the adjustment. Finally, the prediction component predicts the future orientation by integrating all the outputs from the aforementioned components. Let us describe how we predict future orientation in greater detail in the following subsections.

B. ORIENTATION PREDICTION

As the sampling rate of commercial IMUs is very high, *e.g.*, 8 kHz [48], we assume that the direction and velocity of head movement between t and $(t + 1)$ is identical to those between $(t - 1)$ and t. Then, as the orientation displacement can be obtained by multiplying the quaternion derivative $\dot{\mathbf{q}}$ with the sampling period ΔT, the future orientation can be estimated as

$$\widehat{\mathbf{q}}^-_{t+1} = \mathbf{q}_t + \dot{\mathbf{q}}_t \Delta T. \tag{8}$$

However, as mentioned previously, numerical integration of the angular velocity $\boldsymbol{\omega}_t$ induces cumulative errors in $\dot{\mathbf{q}}_t$ because of eventual deviation, significantly degrading the prediction performance. Therefore, inspired by [49] and [50], we remove the error component, which is defined as the relative rotation between the measured inertial direction and the predicted direction of the gravity field, from the measured angular velocity $\boldsymbol{\omega}_t$.

The direction vector of gravity in the Earth frame is given by ${}^E\mathbf{g} = [0, 0, 1]^T$. Then, the direction vector of gravity in the sensor frame ${}^S\mathbf{g}_t$ is obtained by rotating the gravity direction vector ${}^E\mathbf{g}$ with respect to the current orientation $\mathbf{q}_t$ as

$${}^S\mathbf{g}_t = \mathbf{R}(\mathbf{q}_t)^E\mathbf{g}. \tag{9}$$

Thus, the error vector $\mathbf{e}_t$ in the sensor frame can be obtained by [49] and [50]

$$\mathbf{e}_t = \mathbf{a}_t \times {}^S\mathbf{g}_t, \tag{10}$$

where $\mathbf{a}_t$ denotes the measurement vector obtained from the accelerometer. Figure 4 illustrates gravity vectors in the two

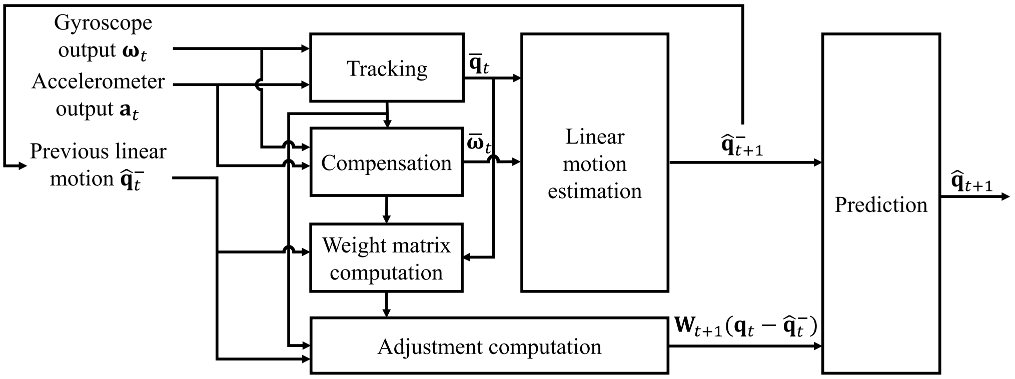

Gyroscope output $\boldsymbol{\omega}_t$
Accelerometer output $\mathbf{a}_t$
Previous linear motion $\widehat{\mathbf{q}}_t^-$
Tracking
$\mathbf{q}_t$
Compensation
$\overline{\boldsymbol{\omega}}_t$
Weight matrix computation
Adjustment computation
Linear motion estimation
$\widehat{\mathbf{q}}_{t+1}^-$
$\mathbf{W}_{t+1}(\mathbf{q}_t - \widehat{\mathbf{q}}_t^-)$
Prediction
$\widehat{\mathbf{q}}_{t+1}$

FIGURE 3. Block diagram of the proposed algorithm for estimating $\widehat{\mathbf{q}}_{t+1}$.

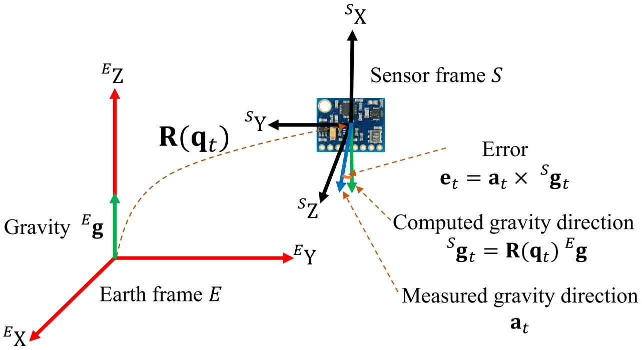

^{S}X
Sensor frame S
^{E}Z
$\mathbf{R}(\mathbf{q}_t)$
^{S}Y
Error
$\mathbf{e}_t = \mathbf{a}_t \times {}^S\mathbf{g}_t$
^{S}Z
Gravity ${}^E\mathbf{g}$
Computed gravity direction
${}^S\mathbf{g}_t = \mathbf{R}(\mathbf{q}_t)\,{}^E\mathbf{g}$
^{E}Y
Earth frame E
Measured gravity direction
$\mathbf{a}_t$
^{E}X

FIGURE 4. Illustration of gravity in the Earth frame E and the sensor frame S.

frames E and S, as well as the error in frame S. Then, the gyroscope output is corrected by compensating for the error as

$$\overline{\boldsymbol{\omega}}_t = \boldsymbol{\omega}_t - K\mathbf{e}_t, \tag{11}$$

where K denotes the correction gain. When $K = 0$, the angular velocity, obtained from the gyroscope, is used directly without compensation. As K increases, more aggressive compensation is made to remove error components from the gyroscope readings. Note that, although the previously estimated orientation $\widehat{\mathbf{q}}_{t-1}$ was used to rotate the gravity direction vector in (9) for tracking in [49] and [50], we use the current true orientation $\mathbf{q}_t$ to ascertain the prediction at the same time t, thereby achieving more accurate prediction of future orientation.

Next, the adjusted rate of change in the orientation $\dot{\mathbf{q}}$ is computed using (3) and the corrected angular velocity in (11) as

$$\dot{\mathbf{q}}_t = \frac{1}{2}\Omega(\overline{\boldsymbol{\omega}}_t)\mathbf{q}_t. \tag{12}$$

Then, the future head orientation in (8) can be rewritten as

$$\widehat{\mathbf{q}}_{t+1}^- = \mathbf{q}_t + \dot{\mathbf{q}}_t\Delta T = \left[\mathbf{I} + \frac{\Delta T}{2}\Omega(\overline{\boldsymbol{\omega}}_t)\right]\mathbf{q}_t = \mathbf{F}_t\mathbf{q}_t, \tag{13}$$

where $\mathbf{F}_t = \mathbf{I} + \frac{\Delta T}{2}\Omega(\overline{\boldsymbol{\omega}}_t)$ is the matrix defining the dynamic system.

Note that the prediction in (13) is based on the assumption of linear motion. Therefore, any nonlinear motion, *e.g.*, abrupt changes in velocity or direction, may degrade the prediction accuracy. Further, prediction of the orientation $\widehat{\mathbf{q}}_{t+1}^-$ in (13) does not account for the prediction accuracy at the previous time instance $(t-1)$, $\widehat{\mathbf{q}}_t^-$. In this work, to improve the accuracy of the current linear prediction $\widehat{\mathbf{q}}_{t+1}^-$, we consider the prediction accuracy of the previous prediction, more specifically, the difference between the current true orientation and the previous linear prediction, $(\mathbf{q}_t - \widehat{\mathbf{q}}_t^-)$. To this end, we define the weight matrix $\mathbf{W} \in \mathbb{R}^{4\times 4}$ to determine the adjustment of the prediction corresponding to a given orientation difference $(\mathbf{q}_t - \widehat{\mathbf{q}}_t^-)$. In this work, the weight matrix is computed similarly to the derivation of the Kalman filter [30]—differences with less uncertainty are assigned greater weights. More specifically, the adjustment is calculated as

$$\mathbf{W}_{t+1}(\mathbf{q}_t - \widehat{\mathbf{q}}_t^-). \tag{14}$$

Then, the future orientation $\widehat{\mathbf{q}}_{t+1}$, calculated using the linear displacement in (13) and the adjustment in (14), can be predicted as

$$\begin{aligned}\widehat{\mathbf{q}}_{t+1} &= \widehat{\mathbf{q}}_{t+1}^- + \mathbf{W}_{t+1}(\mathbf{q}_t - \widehat{\mathbf{q}}_t^-)\\ &= \mathbf{F}_t\mathbf{q}_t + \mathbf{W}_{t+1}(\mathbf{q}_t - \widehat{\mathbf{q}}_t^-).\end{aligned} \tag{15}$$

The weight matrix $\mathbf{W}_{t+1}$ in (15) can be obtained by minimizing the error between the true orientation $\mathbf{q}_{t+1}$ and the predicted orientation $\widehat{\mathbf{q}}_{t+1}$. However, at current time t, information regarding the future orientations $\mathbf{q}_{t+1}$ and $\widehat{\mathbf{q}}_{t+1}$ is unavailable. Therefore, estimating the optimal weight matrix $\mathbf{W}_{t+1}$ is impossible at this time. However, note that, as IMUs can measure samples at a very high sampling rate, the difference between two consecutive samples is very small. This implies that the orientations at time t and $(t+1)$ are similar. Thus, we assume that the weight matrices at time t and $(t+1)$ are similar and use the optimal weight matrix at the previous time instance $\widehat{\mathbf{W}}_t$ for the current prediction, *i.e.*, $\mathbf{W}_{t+1} = \widehat{\mathbf{W}}_t$.

The optimal weight matrix $\widehat{\mathbf{W}}_t$ is determined to minimize the prediction error $\boldsymbol{\epsilon}_t$ in the orientation at time t, which is defined as

$$\boldsymbol{\epsilon}_t = \mathbf{q}_t - \widehat{\mathbf{q}}_t, \tag{16}$$

where $\mathbf{q}_t$ and $\widehat{\mathbf{q}}_t$ denote the true and predicted orientations, respectively. However, note that the true orientation $\mathbf{q}_t$ is unavailable at time instance t. Therefore, instead, we use the tracked orientation $\overline{\mathbf{q}}_t$ to approximate the true orientation. However, as a tracking algorithm is imperfect, the tracked orientation contains observation noise $\mathbf{v}_t$ as

$$\begin{aligned}\mathbf{q}_t &= \overline{\mathbf{q}}_t + \mathbf{v}_t \\ &= \mathbf{F}_t\overline{\mathbf{q}}_{t-1} + \mathbf{v}_t,\end{aligned} \tag{17}$$

where $\mathbf{v}_t$ is zero-mean Gaussian noise with a covariance matrix $\mathbf{C} \in \mathbb{R}^{4\times4}$, *i.e.*, $\mathbf{v}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{C})$. Next, using (15), the predicted orientation $\widehat{\mathbf{q}}_t$ in (16) can be written as the following function of the weight matrix $\mathbf{W}$,

$$\begin{aligned}\widehat{\mathbf{q}}_t &= \mathbf{F}_{t-1}(\overline{\mathbf{q}}_{t-1} + \mathbf{v}_{t-1}) + \mathbf{W}_t(\overline{\mathbf{q}}_{t-1} + \mathbf{v}_{t-1} - \mathbf{F}_{t-2}\overline{\mathbf{q}}_{t-2}) \\ &\simeq \mathbf{F}_{t-1}\overline{\mathbf{q}}_{t-1} + \mathbf{W}_t(\overline{\mathbf{q}}_{t-1} + \mathbf{v}_{t-1} - \mathbf{F}_{t-2}\overline{\mathbf{q}}_{t-2}).\end{aligned} \tag{18}$$

In (18), based on the assumption that both ΔT and $\|\mathbf{v}_{t-1}\|$ are small and $E[\mathbf{v}_{t-1}] = \mathbf{0}$, $\mathbf{F}_{t-1}\mathbf{v}_{t-1}$ is approximated as $\mathbf{F}_{t-1}\mathbf{v}_{t-1} \simeq \mathbf{0}$. Note that, although this approximation is somewhat aggressive, it greatly facilitates the derivation of the closed-form solution, as will be described subsequently. Therefore, the prediction error $\boldsymbol{\epsilon}_t$ in (16) becomes

$$\begin{aligned}\boldsymbol{\epsilon}_t &= (\mathbf{F}_t - \mathbf{F}_{t-1})\overline{\mathbf{q}}_{t-1} - \mathbf{W}_t(\overline{\mathbf{q}}_{t-1} + \mathbf{v}_{t-1} - \mathbf{F}_{t-2}\overline{\mathbf{q}}_{t-2}) \\ &= (\mathbf{F}_t - \mathbf{F}_{t-1})\overline{\mathbf{q}}_{t-1} - \mathbf{W}_t\mathbf{v}_{t-1} - \mathbf{W}_t(\overline{\mathbf{q}}_{t-1} - \mathbf{F}_{t-2}\overline{\mathbf{q}}_{t-2}).\end{aligned} \tag{19}$$

Then, we define the cost function as the mean squared error of the prediction error

$$\begin{aligned}J(\mathbf{W}_t) &= \mathrm{E}\Big[\boldsymbol{\epsilon}_t^T\boldsymbol{\epsilon}_t\Big] \\ &= \mathrm{E}\Big[\mathrm{tr}\Big(\boldsymbol{\epsilon}_t\boldsymbol{\epsilon}_t^T\Big)\Big] \\ &= \mathrm{tr}\Big(\mathrm{E}\Big[\boldsymbol{\epsilon}_t\boldsymbol{\epsilon}_t^T\Big]\Big).\end{aligned} \tag{20}$$

Because $\mathbf{C} = \mathrm{E}\Big[\mathbf{v}_t\mathbf{v}_t^T\Big]$ and $\mathrm{E}[\mathbf{v}_t] = \mathbf{0}$, by the definition of the prediction error $\boldsymbol{\epsilon}$ in (19), we have

$$\begin{aligned}&\mathrm{E}\Big[\boldsymbol{\epsilon}_t\boldsymbol{\epsilon}_t^T\Big] \\ &= [(\mathbf{F}_t - \mathbf{F}_{t-1})\overline{\mathbf{q}}_{t-1}][(\mathbf{F}_t - \mathbf{F}_{t-1})\overline{\mathbf{q}}_{t-1}]^T \\ &\quad - (\mathbf{F}_t - \mathbf{F}_{t-1})\overline{\mathbf{q}}_{t-1}(\overline{\mathbf{q}}_{t-1} - \mathbf{F}_{t-2}\overline{\mathbf{q}}_{t-2})^T\mathbf{W}_t^T \\ &\quad + \mathbf{W}_t\mathbf{C}\mathbf{W}_t^T \\ &\quad - \mathbf{W}_t(\overline{\mathbf{q}}_{t-1} - \mathbf{F}_{t-2}\overline{\mathbf{q}}_{t-2})[(\mathbf{F}_t - \mathbf{F}_{t-1})\overline{\mathbf{q}}_{t-1}]^T \\ &\quad + \mathbf{W}_t(\overline{\mathbf{q}}_{t-1} - \mathbf{F}_{t-2}\overline{\mathbf{q}}_{t-2})(\overline{\mathbf{q}}_{t-1} - \mathbf{F}_{t-2}\overline{\mathbf{q}}_{t-2})^T\mathbf{W}_t^T.\end{aligned} \tag{21}$$

Our objective is to determine the MMSE estimator with the optimal weight matrix $\widehat{\mathbf{W}}_t$ that minimizes $J(\mathbf{W}_t)$ in (20), *i.e.*,

$$\widehat{\mathbf{W}}_t = \underset{\mathbf{W}_t}{\arg\min}\, J(\mathbf{W}_t). \tag{22}$$

TABLE 1. Details of test datasets used for evaluation.

Test datasets	Number of samples	Sample rate
LOPSI [51]	10,000	100 Hz
FKF [47]	27,716	500 Hz
BROAD [52]	30,000	286 Hz
spiralStairs [53]	12,000	256 Hz
straightLine [53]	7,500	
stairsAndCorridor [53]	10,000	

Based on the properties of the trace derivative provided in the Appendix, the partial derivative of $J(\mathbf{W}_t)$ with respect to $\mathbf{W}_t$ is given by

$$\begin{aligned}&\frac{\partial J(\mathbf{W}_t)}{\partial \mathbf{W}_t} \\ &= -2(\mathbf{F}_t - \mathbf{F}_{t-1})\overline{\mathbf{q}}_{t-1}(\overline{\mathbf{q}}_{t-1} - \mathbf{F}_{t-2}\overline{\mathbf{q}}_{t-2})^T \\ &\quad + 2\mathbf{W}_t[(\overline{\mathbf{q}}_{t-1} - \mathbf{F}_{t-2}\overline{\mathbf{q}}_{t-2})(\overline{\mathbf{q}}_{t-1} - \mathbf{F}_{t-2}\overline{\mathbf{q}}_{t-2})^T + \mathbf{C}].\end{aligned} \tag{23}$$

Then, by setting the derivative in (23) to zero, we obtain the optimal weight matrix

$$\begin{aligned}\widehat{\mathbf{W}}_t &= (\mathbf{F}_t - \mathbf{F}_{t-1})\overline{\mathbf{q}}_{t-1}(\overline{\mathbf{q}}_{t-1} - \mathbf{F}_{t-2}\overline{\mathbf{q}}_{t-2})^T \\ &\quad \times[(\overline{\mathbf{q}}_{t-1} - \mathbf{F}_{t-2}\overline{\mathbf{q}}_{t-2})(\overline{\mathbf{q}}_{t-1} - \mathbf{F}_{t-2}\overline{\mathbf{q}}_{t-2})^T + \mathbf{C}]^{-1} \\ &= (\overline{\mathbf{q}}_t - \widehat{\mathbf{q}}_t^-)(\overline{\mathbf{q}}_{t-1} - \widehat{\mathbf{q}}_{t-1}^-)^T \\ &\quad \times[(\overline{\mathbf{q}}_{t-1} - \widehat{\mathbf{q}}_{t-1}^-)(\overline{\mathbf{q}}_{t-1} - \widehat{\mathbf{q}}_{t-1}^-)^T + \mathbf{C}]^{-1}.\end{aligned} \tag{24}$$

Finally, assuming that the observation noise is negligible and substituting $\mathbf{q}_t$ in (17) and $\widehat{\mathbf{W}}_t$ in (24) into (15), we obtain the optimally predicted future orientation as

$$\begin{aligned}\widehat{\mathbf{q}}_{t+1} &= \mathbf{F}_t\overline{\mathbf{q}}_t + (\overline{\mathbf{q}}_t - \widehat{\mathbf{q}}_t^-)(\overline{\mathbf{q}}_{t-1} - \widehat{\mathbf{q}}_{t-1}^-)^T \\ &\quad \times[(\overline{\mathbf{q}}_{t-1} - \widehat{\mathbf{q}}_{t-1}^-)(\overline{\mathbf{q}}_{t-1} - \widehat{\mathbf{q}}_{t-1}^-)^T + \mathbf{C}]^{-1}(\overline{\mathbf{q}}_t - \widehat{\mathbf{q}}_t^-).\end{aligned} \tag{25}$$

IV. EXPERIMENTAL RESULTS

We evaluate the performance of the proposed orientation prediction algorithm in comparison with those of five conventional prediction algorithms: EKF, UKF, and three of their extensions, *i.e.*, covariance inflated-multiplicative extended Kalman filter (CI-MEKF) [36], decoupled orientation estimation approach (DOEA) [38], and robust adaptive unscented Kalman filter (RAUKF) [40]. For reproducibility, we provide the source codes on our project website.[1]

A. EXPERIMENTAL SETTINGS

Six real-world test datasets with different sampling frequencies and types of movements are used for evaluation—three test datasets with ground-truth orientations (LOPSI [51], FKF [47], and BROAD [52]) and three test datasets without ground-truth orientations [53] (spiralStairs, straightLine, and stairsAndCorridor). The LOPSI dataset is collected using a foot-mounted IMU, which contains sensor data of a straight trajectory at a sampling rate of 100 Hz. The FKF test dataset

[1] https://github.com/hueledao/Orientation_Prediction

includes motion randomly generated using a human hand at a sampling rate of 500 Hz, which is the highest sampling rate in the test datasets considered in this paper. BROAD is a benchmark for robust inertial orientation estimation, collected by rotating an IMU with measurements recorded at a sampling rate of 286 Hz. All datasets without ground-truth orientations [53] are obtained from IMUs attached to feet and contain measurements of different types of motion recorded at a sampling rate of 256 Hz. All the datasets contain seven components $[\omega_x, \omega_y, \omega_z, a_x, a_y, a_z, \Delta T]^T$, where ω and a denote the x-, y-, and z-components of gyroscope and accelerometer readings, respectively, at each time instance, and ΔT denotes the sampling period. Further details of test datasets are listed in Table 1.

The parameter K in (11) is fixed to 0.5 as recommended in [50], and the observation uncertainty $\mathbf{C}$ in (24) is set to $\mathbf{C} = 10^{-4} \times \mathbf{I}$, unless otherwise specified, to achieve the best overall performance. The predicted orientations are evaluated in Euler angles as in [42] to facilitate comparison and visualization. We evaluate the prediction performance by comparing the root-mean-square error (RMSE) between the predicted and ground-truth orientations on test datasets with ground-truth orientations and that between predicted orientations and tracked orientations on test datasets without ground-truth orientations. The RMSE error is defined as

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\|\mathbf{u}_i - \widehat{\mathbf{u}}_i\|^2}, \qquad (26)$$

where $\widehat{\mathbf{u}} = [\hat{u}^r, \hat{u}^p, \hat{u}^y]^T$ and $\mathbf{u} = [u^r, u^p, u^y]^T$ denote the predicted and ground-truth (or tracked) Euler angles, *i.e.*, roll, pitch, and yaw, respectively, and N is the number of samples in each test dataset.

B. QUANTITATIVE EVALUATION

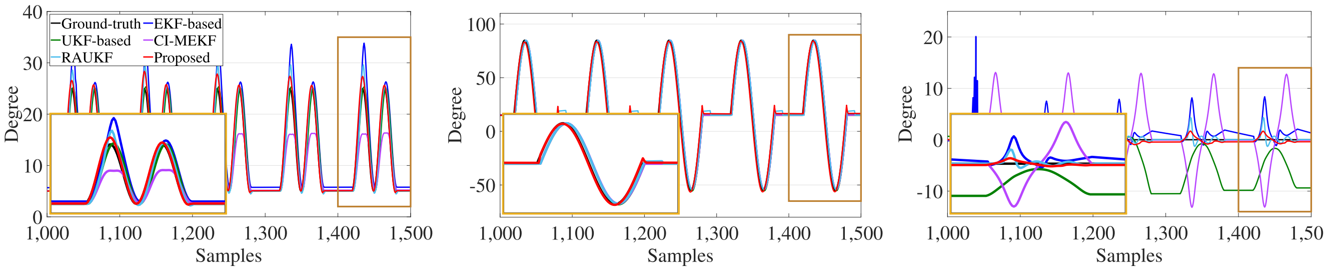
Table 2 compares the prediction performance on the test datasets with ground-truth orientations. The EKF algorithm yields the worst results in all cases. This can be attributed to the use of the output of only two sensors (not the magnetometer output) and linearization of state transition functions of the current estimation to predict future orientations. The UKF algorithm uses unscented transformation instead of linearization to improve the prediction performance. However, the yaw angles predicted by UKF remain inaccurate, as it also does not use a magnetometer to gain information on yaw angles. CI-MEKF, DOEA, and RAUKF use additional information from a magnetometer; thus, they yield better yaw angle predictions than EKF and UKF. However, as magnetic disturbance negatively affects the roll and pitch predictions, they exhibit lower accuracy in the roll and pitch predictions than EKF and UKF. For example, on the LOPSI test dataset, CI-MEKF and DOEA predict the yaw angle with lower errors than EKF but exhibits higher prediction errors for the roll and pitch angles. RAUKF exhibits better performance than CI-MEKF and DOEA. In particular, RAUKF yields significantly lower prediction errors than CI-MEKF and DOEA in yaw. Finally, the proposed algorithm exhibits the best overall prediction errors because it leverages the accuracy of previous predictions and utilizes tracked orientation instead of predicted orientation, unlike the conventional algorithms. Nevertheless, note that, since the proposed algorithm uses only two sensors, whereas RAUKF uses three sensors, the proposed algorithm yields higher prediction errors for yaw angle on the FKF dataset. In addition, because the BROAD dataset contains significant fluctuations in roll and pitch, the proposed algorithm that uses the accuracy of the previous predictions yields relatively large errors.

TABLE 2. Comparison of RMSE values for euler angles (°) between prediction and ground-truth orientations. For each euler angle, The boldface numbers denote the best results for each test dataset.

Test datasets	Algorithms	Roll	Pitch	Yaw	Overall
LOPSI	EKF	2.16	3.57	6.22	4.32
	UKF	1.09	3.78	4.05	3.26
	CI-MEKF [36]	2.79	4.22	3.94	3.70
	DOEA [38]	3.66	3.77	2.11	3.27
	RAUKF [40]	0.92	3.51	0.85	2.15
	Proposed	**0.87**	**2.01**	**0.72**	**1.33**
FKF	EKF	6.28	2.69	5.10	4.92
	UKF	2.43	1.39	5.25	3.44
	CI-MEKF [36]	5.92	2.52	4.86	4.66
	DOEA [38]	4.89	2.97	3.54	3.88
	RAUKF [40]	2.37	1.16	**1.76**	1.83
	Proposed	**1.59**	**0.97**	2.14	**1.64**
BROAD	EKF	4.95	2.27	14.98	9.20
	UKF	4.23	1.68	10.13	6.41
	CI-MEKF [36]	1.95	1.10	8.18	4.90
	DOEA [38]	**0.68**	2.07	1.98	1.70
	RAUKF [40]	2.43	**1.02**	1.09	1.65
	Proposed	2.11	1.40	**0.85**	**1.54**

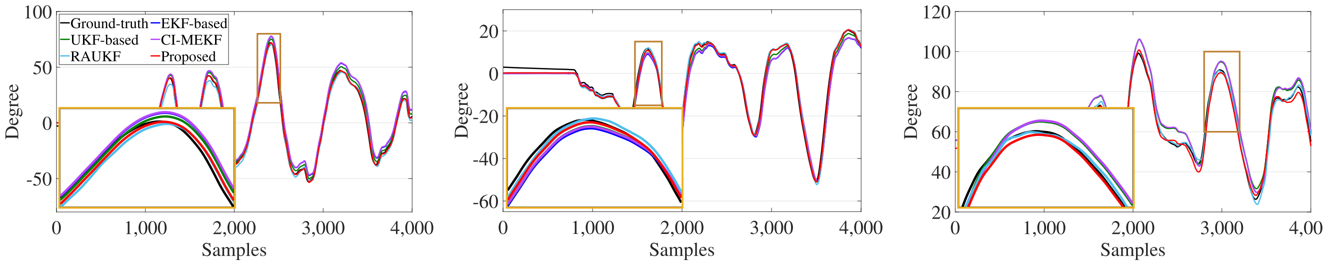
TABLE 3. Comparison of RMSE values for the euler angles (°) between predicted and tracked orientations. For each Euler angle, The boldface numbers denote the best results for each test dataset.

Test datasets	Algorithms	Roll	Pitch	Yaw	Overall
spiralStairs	EKF	1.96	2.26	9.33	5.66
	UKF	2.73	3.12	5.77	4.10
	CI-MEKF [36]	2.12	2.56	6.71	4.32
	DOEA [38]	2.71	2.67	3.83	3.12
	RAUKF [40]	2.75	2.99	2.38	2.72
	Proposed	**0.14**	**0.13**	**0.06**	**0.11**
straightLine	EKF	1.62	2.59	4.98	3.37
	UKF	1.28	3.66	4.19	3.30
	CI-MEKF [36]	2.79	2.83	3.07	2.90
	DOEA [38]	1.81	2.75	3.06	2.60
	RAUKF [40]	2.08	4.00	2.54	2.99
	Proposed	**0.09**	**0.13**	**0.08**	**0.10**
stairsAndCorridor	EKF	1.66	2.54	10.77	6.46
	UKF	1.52	2.39	10.14	6.08
	CI-MEKF [36]	1.60	2.61	8.97	5.48
	DOEA [38]	2.51	2.91	7.85	5.05
	RAUKF [40]	1.95	3.26	5.44	3.83
	Proposed	**0.17**	**0.16**	**3.60**	**2.08**



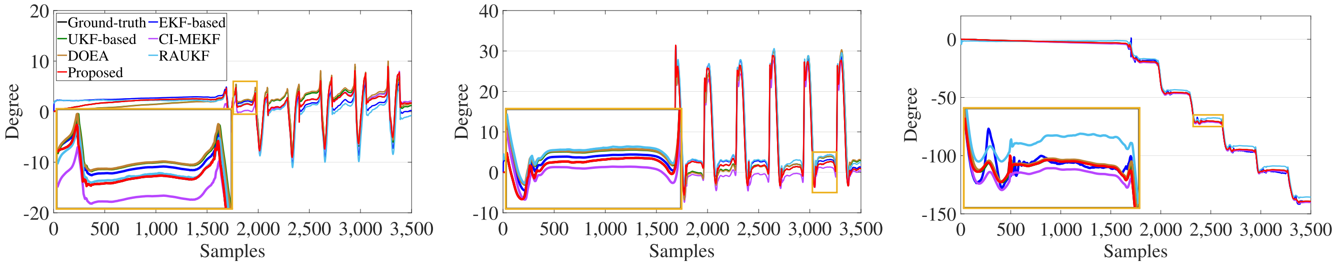
(a) Roll (b) Pitch (c) Yaw

FIGURE 5. Comparison of the predicted Euler angles on the LOPSI test dataset.



(a) Roll (b) Pitch (c) Yaw

FIGURE 6. Comparison of the predicted euler angles on the FKF test dataset.

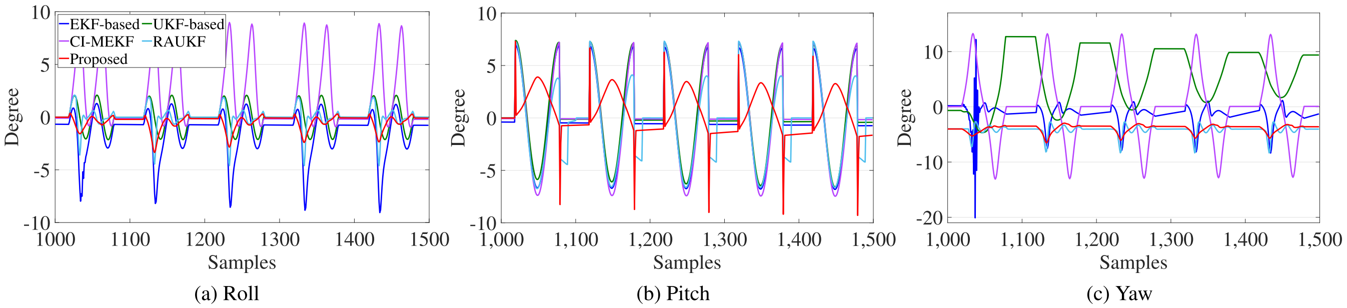


(a) Roll (b) Pitch (c) Yaw

FIGURE 7. Comparison of the predicted Euler angles on the spiralStairs test dataset.

Table 3 lists the RMSE between the predicted and tracked Euler angles on test datasets without ground-truth orientations. As in the previous case, EKF yields the highest errors on all test datasets, especially for yaw prediction, as it does not use magnetometer outputs. CI-MEKF, DOEA, and RAUKF exhibit higher overall prediction accuracy than EKF and UKF as well, especially for yaw prediction, as they use more sensors. However, their roll and pitch predictions are worse than those of EKF and UKF because of the magnetic disturbances that affect the measurement of the roll and pitch angles. The proposed algorithm exhibits significantly higher prediction accuracy for all Euler angles without exception—its RMSE values are significantly smaller than those of the conventional algorithms. This is because the proposed algorithm predicts future orientation based on the current tracked orientation, whereas the conventional algorithms do the same based on previous predictions of orientation.

Figures 5–7 plot the Euler angles predicted by each algorithm and the ground-truth on selected datasets. To facilitate the analysis, only a part of samples is plotted. Figures 5(a)–(c) compare predictions of the roll, pitch, and yaw, respectively, on the LOPSI test dataset with the zoomed insets corresponding to the yellow boxes. As the proposed algorithm uses tracked orientation for prediction, its predictions of the roll and pitch angles are closer to the ground-truth, as depicted in Figures 5(a) and (b). The conventional algorithms exhibit larger errors for these angles. Although, the roll trajectories of UKF and RAUKF and the pitch predictions of all conventional algorithms are similar to the ground-truth, certain deviations are observed as these algorithms utilize previous predictions of orientations unlike the proposed algorithm. Figure 5(c) demonstrates that the proposed algorithm exhibits the lowest yaw prediction error, where the LOPSI test dataset does not contain yaw rotation. This indicates that, although information from only two sensors is used, the proposed



(a) Roll (b) Pitch (c) Yaw

FIGURE 8. Comparison of euler angle prediction errors on the LOPSI test dataset.

algorithm effectively avoids large errors by incorporating the accuracy of previous predictions into the prediction process.

Figures 6 and 7 compare roll, pitch, and yaw predictions on the FKF and spiralStairs datasets, respectively. The results reveal tendencies similar to those in Figure 5. In particular, as depicted in Figure 6, the proposed algorithm predicts all Euler angles with significantly higher accuracy than conventional algorithms. The EKF-based algorithms, *i.e.*, EKF, CI-MEKF, and DOEA, yield larger errors on this test dataset; their curves exhibit significant deviation from the ground-truth. CI-MEKF, DOEA, and RAUKF outperform EKF and UKF in terms of all Euler angles. Predictions of RAUKF and the proposed algorithm agree with the ground-truth orientations. Figure 7 also indicates that the predictions obtained using the proposed algorithm are in close agreement with the tracking orientations—the curve corresponding to the proposed algorithm (red curve) and the ground-truth curve (black curve) are almost identical.

To facilitate the evaluation, Figures 8–10 visualize the prediction errors computed as the difference between the ground-truth (or tracked) orientations and predicted orientations on the LOPSI, FKF, and spiralStairs test datasets, respectively. The roll and pitch errors in Figures 8–9(a) and (b), respectively, demonstrate that the prediction errors of the proposed algorithm are the smallest, indicating the most accurate prediction of orientations. In Figure 8(b), the pitch error of the proposed algorithm is high only corresponding to samples with rapid changes in movement. In Figure 9, RAUKF and the proposed algorithm exhibit almost identical errors and error ranges, whereas the other algorithms exhibit higher errors for all Euler angles. Finally, Figure 10 compares the prediction errors on the spiralStairs dataset. As the proposed algorithm uses tracked orientations rather than previously predicted orientation during the prediction process, it also exhibits the smallest errors for all Euler angles.

C. EFFECTIVENESS OF WEIGHT MATRIX W

As discussed in Section III-B, the weight matrix in (24) determines the adjustments of the prediction based on the accuracy of previous predictions to improve the prediction accuracy. We evaluate the effects of the weight matrix on the prediction performance. Table 4 compares the prediction performance for various configurations of the weight matrix $\mathbf{W}$. First, $\mathbf{W} = \mathbf{0}$, which corresponds to linear prediction, yields the worst overall prediction performance. This is because previous prediction accuracy is not taken into account during prediction. Second, fixed weight matrix yields better prediction performance by adjusting linear prediction based on the prediction accuracy of the previous prediction. However, as the prediction performance is affected by the characteristics of test sets, it is difficult to obtain the optimal weight matrix. For example, whereas $\mathbf{W} = 10^{-2} \times \mathbf{I}$ yields the lowest overall error on LOPSI, $\mathbf{W} = 10^{-1} \times \mathbf{I}$ is optimal for spiralStairs. Finally, the proposed algorithm, which adaptively estimates $\mathbf{W}$ in an MMSE sense to minimize the expected prediction error, yields the best overall prediction performance.

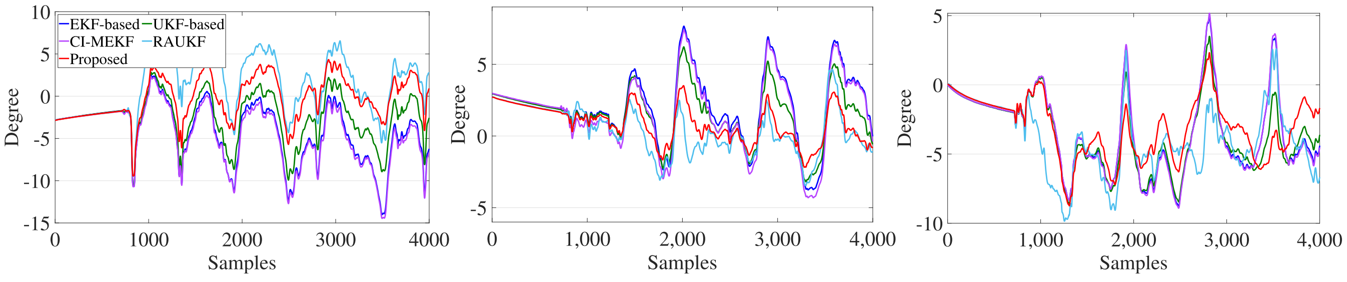
D. IMPACTS OF COVARIANCE MATRIX C

We analyze the effects of the observation uncertainty $\mathbf{C}$ in (24), which is the covariance matrix of the observation noise $\mathbf{v}_t$ in (17), to the prediction performance. Table 5 compares the prediction performances of the proposed algorithm for different values of $\mathbf{C}$ on the LOPSI and spiralStairs test datasets. Although different values of $\mathbf{C}$ yield the minimum roll, pitch, and yaw angle errors, $\mathbf{C} = 10^{-4} \times \mathbf{I}$ provides the lowest overall errors for both datasets. Therefore, to achieve the overall best prediction performance, we fixed $\mathbf{C} = 10^{-4} \times \mathbf{I}$ in this work.

E. COMPUTATION TIMES

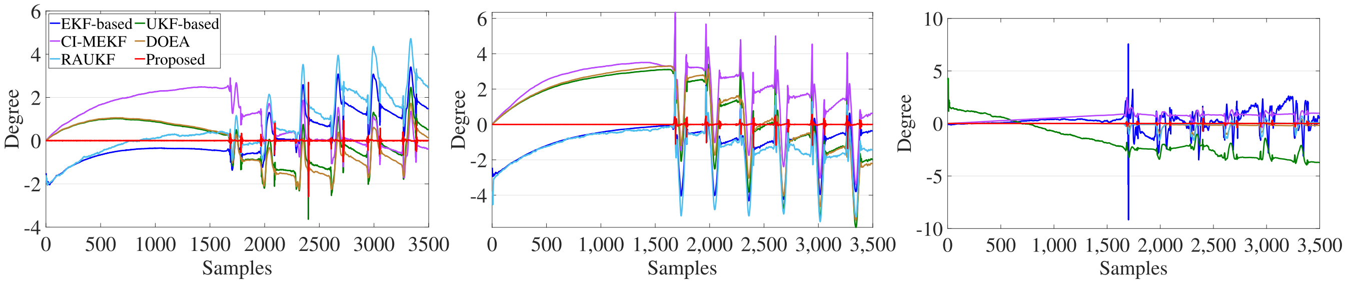
Table 6 compares the average execution times obtained by different algorithms over 100 independent runs on the LOPSI dataset. In this test, we use MATLAB version R2021a on a computer with a 3.60 GHz CPU and 32 GB RAM.

Although EKF is the most efficient in terms of computation time, it exhibits the highest error, as shown in Tables 2 and 3. As mentioned previously, UKF is more complex than EKF. As CI-MEKF, DOEA, and RAUKF use additional information from a magnetometer, they require significantly longer execution times than their counterparts, *i.e.*, EKF and UKF. Although the proposed algorithm is more



(a) Roll (b) Pitch (c) Yaw

FIGURE 9. Comparison of euler angle prediction errors on the FKF test dataset.



(a) Roll (b) Pitch (c) Yaw

FIGURE 10. Comparison of euler angle prediction errors on the spiralstairs test dataset.

TABLE 4. Prediction performance evaluation for various settings of weight matrix *W* in terms of RMSE values for the Euler angles (°).

Test datasets	W	Roll	Pitch	Yaw	Overall
LOPSI	$\mathbf{W} = \mathbf{0}$	0.9937	1.9842	0.8055	1.3630
	$\mathbf{W} = 10^{-1} \times \mathbf{I}$	**0.8350**	2.0087	0.7364	1.3503
	$\mathbf{W} = 10^{-2} \times \mathbf{I}$	0.9647	**1.9752**	0.7799	1.3466
	$\mathbf{W} = 10^{-3} \times \mathbf{I}$	0.9676	1.9754	0.7818	1.3478
	$\mathbf{W} = 10^{-4} \times \mathbf{I}$	0.9679	1.9754	0.7820	1.3479
	Proposed	0.8663	2.0087	**0.7214**	**1.3299**
spiralStairs	$\mathbf{W} = \mathbf{0}$	0.1325	0.1386	0.0804	0.1200
	$\mathbf{W} = 10^{-1} \times \mathbf{I}$	**0.1306**	0.1300	0.0748	0.1148
	$\mathbf{W} = 10^{-2} \times \mathbf{I}$	0.1323	0.1377	0.0798	0.1195
	$\mathbf{W} = 10^{-3} \times \mathbf{I}$	0.1325	0.1385	0.0803	0.1200
	$\mathbf{W} = 10^{-4} \times \mathbf{I}$	0.1325	0.1385	0.0804	0.1200
	Proposed	0.1359	**0.1274**	**0.0623**	**0.1134**

TABLE 5. Prediction performance evaluation for different covariance matrices **C** in terms of RMSE values for the Euler angles (°).

Test datasets	C	Roll	Pitch	Yaw	Overall
LOPSI	$10^{-2} \times \mathbf{I}$	0.9664	**1.9756**	0.7811	1.3475
	$10^{-3} \times \mathbf{I}$	0.9536	1.9777	0.7735	1.3440
	$10^{-4} \times \mathbf{I}$	0.8663	2.0087	0.7214	**1.3299**
	$10^{-5} \times \mathbf{I}$	0.7072	2.2673	0.6221	1.4175
	$10^{-6} \times \mathbf{I}$	**0.6616**	2.6309	**0.5911**	1.6030
spiralStairs	$10^{-2} \times \mathbf{I}$	0.1322	0.1382	0.0796	0.1196
	$10^{-3} \times \mathbf{I}$	**0.1305**	0.1356	0.0742	0.1168
	$10^{-4} \times \mathbf{I}$	0.1359	**0.1274**	0.0623	**0.1134**
	$10^{-5} \times \mathbf{I}$	0.1887	0.1303	**0.0604**	0.1369
	$10^{-6} \times \mathbf{I}$	0.2352	0.1517	0.0663	0.1661

complex than EKF and requires longer computation time, it exhibits higher accuracy compared to existing algorithms.

TABLE 6. Average execution time per sample in microseconds of each algorithm over 100 independent runs.

EKF	UKF	CI-MEKF [36]	DOEA [38]	RAUKF [40]	Proposed
10.88	245.66	60.35	46.97	1,993.63	21.69

V. CONCLUSION

In this work, we proposed an orientation prediction algorithm for VR and AR devices equipped with an IMU based on Kalman-like error compensation. The main contribution of this work is the improvement of the the current prediction accuracy based on the previous prediction accuracy and the determination of the optimal adjustment for predictions based on the orientation difference. To this end, we first estimated initial future orientations based on the assumption of linear movement. Then, we adjusted the orientation displacement by considering a weighted difference between the current orientation and the previous linear prediction. The weights were computed by considering the estimation errors based on the MMSE criterion using Kalman-like error compensation. Experimental results demonstrated that the proposed algorithm exhibits higher orientation prediction accuracy for all Euler angles than conventional algorithms on several real-world test datasets.

APPENDIX

The derivatives of traces of matrices [54] used in this paper are

$$\frac{\partial}{\partial \mathbf{X}} \mathrm{tr}(\mathbf{A}\mathbf{X}^T) = \mathbf{A} \tag{27}$$

$$\frac{\partial}{\partial \mathbf{X}} \mathrm{tr}(\mathbf{XA}) = \mathbf{A}^T. \tag{28}$$

Further, if **A** is symmetric, then

$$\frac{\partial}{\partial \mathbf{X}} \mathrm{tr}(\mathbf{XAX}^T) = 2\mathbf{XA}. \tag{29}$$

REFERENCES

[1] D. W. F. van Krevelen and R. Poelman, "A survey of augmented reality technologies, applications and limitations," *Int. J. Virtual Reality*, vol. 9, no. 2, pp. 1–20, Jun. 2010.

[2] L. P. Berg and J. M. Vance, "Industry use of virtual reality in product design and manufacturing: A survey," *Virtual Reality*, vol. 21, no. 1, pp. 1–17, Mar. 2017.

[3] G. Bhorkar, "A survey of augmented reality navigation," 2017, *arXiv:1708.05006*.

[4] M. K. Bekele, R. Pierdicca, E. Frontoni, E. S. Malinverni, and J. Gain, "A survey of augmented, virtual, and mixed reality for cultural heritage," *J. Comput. Cultural Heritage*, vol. 11, no. 2, pp. 1–36, Jun. 2018.

[5] K. Kim, M. Billinghurst, G. Bruder, H. B.-L. Duh, and G. F. Welch, "Revisiting trends in augmented reality research: A review of the 2 nd decade of ISMAR (2008–2017)," *IEEE Trans. Vis. Comput. Graph.*, vol. 24, no. 11, pp. 2947–2962, Nov. 2018.

[6] L. F. de Souza Cardoso, F. C. M. Q. Mariano, and E. R. Zorzal, "A survey of industrial augmented reality," *Comput. Ind. Eng.*, vol. 139, Jan. 2020, Art. no. 106159.

[7] E. Bastug, M. Bennis, M. Médard, and M. Debbah, "Toward interconnected virtual reality: Opportunities, challenges, and enablers," *IEEE Commun. Mag.*, vol. 55, no. 6, pp. 110–117, Jun. 2017.

[8] G. Samaraweera, R. Guo, and J. Quarles, "Head tracking latency in virtual environments revisited: Do users with multiple sclerosis notice latency less?" *IEEE Trans. Vis. Comput. Graph.*, vol. 22, no. 5, pp. 1630–1636, May 2016.

[9] M. Yang, J. Zhang, and L. Yu, "Perceptual tolerance to motion-to-photon latency with head movement in virtual reality," in *Proc. Picture Coding Symp.*, Nov. 2019, pp. 1–5.

[10] A. Battiston, I. Sharf, and M. Nahon, "Attitude estimation for collision recovery of a quadcopter unmanned aerial vehicle," *Int. J. Robot. Res.*, vol. 38, nos. 10–11, pp. 1286–1306, Aug. 2019.

[11] S. Wilson, H. Eberle, Y. Hayashi, S. O. H. Madgwick, A. McGregor, X. Jing, and R. Vaidyanathan, "Formulation of a new gradient descent MARG orientation algorithm: Case study on robot teleoperation," *Mech. Syst. Signal Process.*, vol. 130, pp. 183–200, Sep. 2019.

[12] Á. Odry, R. Fullér, I. J. Rudas, and P. Odry, "Kalman filter for mobile-robot attitude estimation: Novel optimized and adaptive solutions," *Mech. Syst. Signal Process.*, vol. 110, pp. 569–589, Sep. 2018.

[13] Y. Tian, W. R. Hamel, and J. Tan, "Accurate human navigation using wearable monocular visual and inertial sensors," *IEEE Trans. Instrum. Meas.*, vol. 63, no. 1, pp. 203–213, Jan. 2014.

[14] X. Xu, X. Xu, T. Zhang, and Z. Wang, "In-motion filter-QUEST alignment for strapdown inertial navigation systems," *IEEE Trans. Instrum. Meas.*, vol. 67, no. 8, pp. 1979–1993, Aug. 2018.

[15] Z. Qiu, Y. Huang, and H. Qian, "Adaptive robust nonlinear filtering for spacecraft attitude estimation based on additive quaternion," *IEEE Trans. Instrum. Meas.*, vol. 69, no. 1, pp. 100–108, Jan. 2020.

[16] J. D. Hol, T. B. Schon, F. Gustafsson, and P. J. Slycke, "Sensor fusion for augmented reality," in *Proc. Int. Conf. Inf. Fusion*, Jul. 2006, pp. 1–6.

[17] T. Lupton and S. Sukkarieh, "Visual-inertial-aided navigation for high-dynamic motion in built environments without initial conditions," *IEEE Trans. Robot.*, vol. 28, no. 1, pp. 61–76, Feb. 2012.

[18] N. S. Lakshmprabha, S. Kasderidis, P. Mousouliotis, L. Petrou, and O. Beltramello, "Augmented reality for maintenance application on a mobile platform," in *Proc. IEEE Virtual Reality*, Mar. 2015, pp. 355–356.

[19] L. Schaffer, Z. Kincses, and S. Pletl, "A real-time pose estimation algorithm based on FPGA and sensor fusion," in *Proc. IEEE Int. Symp. Intell. Syst. Informat.*, Sep. 2018, pp. 149–154.

[20] J. He, C. Sun, B. Zhang, and P. Wang, "Adaptive error-state Kalman filter for attitude determination on a moving platform," *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1–10, 2021.

[21] P. Corke, J. Lobo, and J. Dias, "An introduction to inertial and visual sensing," *Int. J. Robot. Res.*, vol. 26, no. 6, pp. 519–535, 2007.

[22] M. Kok, J. D. Hol, and T. B. Schön, "Using inertial sensors for position and orientation estimation," *Found. Trends Signal Process.*, vol. 11, nos. 1–2, pp. 1–153, 2017.

[23] E. Foxlin, "Motion tracking requirements and technologies," *Handbook Virtual Environ. Technol.*, vol. 8, pp. 163–210, Jan. 2002.

[24] D. Roetenberg, H. J. Luinge, C. T. M. Baten, and P. H. Veltink, "Compensation of magnetic disturbances improves inertial and magnetic sensing of human body segment orientation," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 13, no. 3, pp. 395–405, Sep. 2005.

[25] Y. Akatsuka and G. A. Bekey, "Compensation for end to end delays in a VR system," in *Proc. IEEE Virtual Reality Annu. Int. Symp.*, Mar. 1998, pp. 156–159.

[26] A. Kiruluta, M. Eizenman, and S. Pasupathy, "Predictive head movement tracking using a Kalman filter," *IEEE Trans. Syst., Man, B, Cybern.*, vol. 27, no. 2, pp. 326–331, Apr. 1997.

[27] F. Ababsa, J.-Y. Didier, M. Mallem, and D. Roussel, "Head motion prediction in augmented reality systems using Monte Carlo particle filters," in *Proc. Int. Cof. Artif. Reality Telexixtence*, Dec. 2003, pp. 83–88.

[28] R. G. Valenti, I. Dryanovski, and J. Xiao, "A linear Kalman filter for MARG orientation estimation using the algebraic quaternion algorithm," *IEEE Trans. Instrum. Meas.*, vol. 65, no. 2, pp. 467–481, Feb. 2016.

[29] H. Ahmed and M. Tahir, "Improving the accuracy of human body orientation estimation with wearable IMU sensors," *IEEE Trans. Instrum. Meas.*, vol. 66, no. 3, pp. 535–542, Mar. 2017.

[30] R. Faragher, "Understanding the basis of the Kalman filter via a simple and intuitive derivation," *IEEE Signal Process. Mag.*, vol. 29, no. 5, pp. 128–132, Sep. 2012.

[31] A. van Rhijn, R. van Liere, and J. D. Mulder, "An analysis of orientation prediction and filtering methods for VR/AR," in *Proc. IEEE Virtual Reality*, Mar. 2005, pp. 67–74.

[32] F. Ababsa, M. Mallem, and D. Roussel, "Comparison between particle filter approach and Kalman filter-based technique for head tracking in augmented reality systems," in *Proc. IEEE Int. Conf. Robot. Autom.*, Apr. 2004, pp. 1021–1026.

[33] A. Assa and F. Janabi-Sharifi, "A Kalman filter-based framework for enhanced sensor fusion," *IEEE Sensors J.*, vol. 15, no. 6, pp. 3281–3292, Jun. 2015.

[34] H. Zhao and Z. Wang, "Motion measurement using inertial sensors, ultrasonic sensors, and magnetometers with extended Kalman filter for data fusion," *IEEE Sensors J.*, vol. 12, no. 5, pp. 943–953, May 2012.

[35] X. Tong, Z. Li, G. Han, N. Liu, Y. Su, J. Ning, and F. Yang, "Adaptive EKF based on HMM recognizer for attitude estimation using MEMS MARG sensors," *IEEE Sensors J.*, vol. 18, no. 8, pp. 3299–3310, Apr. 2018.

[36] 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 Trans. Instrum. Meas.*, vol. 67, no. 1, pp. 209–217, Jan. 2018.

[37] S. Park, J. Park, and C. G. Park, "Adaptive attitude estimation for low-cost MEMS IMU using ellipsoidal method," *IEEE Trans. Instrum. Meas.*, vol. 69, no. 9, pp. 7082–7091, Sep. 2020.

[38] Y. Sun, X. Xu, X. Tian, L. Zhou, and Y. Li, "A decoupled orientation estimation approach for robust roll and pitch measurements in magnetically disturbed environment," *IEEE Trans. Instrum. Meas.*, vol. 71, pp. 1–11, 2022.

[39] H. G. de Marina, F. J. Pereda, J. M. Giron-Sierra, and F. Espinosa, "UAV attitude estimation using unscented Kalman filter and TRIAD," *IEEE Trans. Ind. Electron.*, vol. 59, no. 11, pp. 4465–4474, Nov. 2012.

[40] A. C. B. Chiella, B. O. S. Teixeira, and G. A. S. Pereira, "Robust attitude estimation using an adaptive unscented Kalman filter," in *Proc. Int. Conf. Robot. Autom.*, May 2019, pp. 7748–7754.

[41] X. Tong, Y. Su, Z. Li, C. Si, G. Han, J. Ning, and F. Yang, "A double-step unscented Kalman filter and HMM-based zero-velocity update for pedestrian dead reckoning using MEMS sensors," *IEEE Trans. Ind. Electron.*, vol. 67, no. 1, pp. 581–591, Jan. 2020.

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

[43] M. D. Shuster, "A survey of attitude representations," *Navigation*, vol. 8, no. 9, pp. 439–517, Oct. 1993.

[44] I. Y. Bar-Itzhack and Y. Oshman, "Attitude determination from vector observations: Quaternion estimation," *IEEE Trans. Aerosp. Electron. Syst.*, vol. AES-21, no. 1, pp. 128–136, Jan. 1985.

[45] M.-W. Seo, S.-W. Choi, and S.-J. Kang, "Sensor-based latency reduction method for virtual reality head mounted display," in *SID Symp. Dig. Tech. Papers*, May 2017, vol. 48, no. 1, pp. 1641–1644.

[46] H. J. Luinge and P. H. Veltink, "Measuring orientation of human body segments using miniature gyroscopes and accelerometers," *Med. Biol. Eng. Comput.*, vol. 43, no. 2, pp. 273–282, Mar. 2005.

[47] S. Guo, J. Wu, Z. Wang, and J. Qian, "Novel MARG-sensor orientation estimation algorithm using fast Kalman filter," *J. Sensors*, vol. 2017, pp. 1–12, Jan. 2017.

[48] G. Coviello and G. Avitabile, "Multiple synchronized inertial measurement unit sensor boards platform for activity monitoring," *IEEE Sensors J.*, vol. 20, no. 15, pp. 8771–8777, Aug. 2020.

[49] 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.

[50] 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.

[51] LOPSI. *Foot Mounted IMU Data Sets for the Evaluation of PDR Algorithms*. Accessed: Oct. 29, 2022. [Online]. Available: http://lopsi.weebly.com/downloads.html

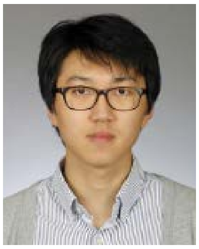
[52] D. Laidig, M. Caruso, A. Cereatti, and T. Seel, "BROAD—A benchmark for robust inertial orientation estimation," *Data*, vol. 6, no. 7, p. 72, Jun. 2021.

[53] 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.*, Jun. 2011, pp. 1–7.

[54] K. B. Petersen and M. S. Pedersen, "The matrix cookbook," Tech. Univ. Denmark, Lyngby, Denmark, Tech. Rep. IMM2012-03274, Nov. 2012.

LE THI HUE DAO (Student Member, IEEE) received the B.S. degree in information technology from the University of Transport and Communications, Hanoi, Vietnam, in 2016, and the M.S. degree in electronics and telecommunication from the University of Engineering and Technology, Vietnam National University, Hanoi, in 2019. She is currently pursuing the Ph.D. degree with the Department of Multimedia Engineering, Dongguk University, Seoul, South Korea. Her research interests include pose estimation for VR and AR devices.

TRUONG THANH NHAT MAI (Student Member, IEEE) received the B.S. degree in mathematics and computer science from the Ho Chi Minh City University of Science, Ho Chi Minh City, Vietnam, in 2014, and the M.S. degree in electrical, electronic, and control engineering from Hankyong National University, Anseong, South Korea, in 2017. He is currently pursuing the Ph.D. degree with the Department of Multimedia Engineering, Dongguk University, Seoul, South Korea. His research interests include image processing and computational imaging.



WOOK HONG received the B.S. and M.S. degrees in electrical and electronics engineering from Kangwon National University, Chuncheon, South Korea, in 2013 and 2015, respectively. Since 2015, he has been working as a Digital Circuit Design Engineer with RAONTECH, Seongnam, South Korea, where he is involved in designing display controllers for liquid crystal on silicon microdisplays.

SANGHYUN PARK received the B.S. and M.S. degrees in electronic engineering from Korea University, Seoul, South Korea, in 2017 and 2019, respectively. He is currently a Senior Engineer with the Research and Development Group, RAONTECH. His research interests include RISC processor architecture and its hardware/software optimization for microdisplay systems.

HOKWON KIM received the B.S. degree in electronic engineering from Korea University, Seoul, South Korea, in 2005. He is currently the Principal Digital Engineer with the Research and Development Group, RAONTECH. He has been designed digital circuits for various microdisplay backplanes including OLED on silicon and microLED on silicon microdisplays.

JOON GOO LEE received the B.S., M.S., and Ph.D. degrees in electrical engineering from Korea University, Seoul, South Korea, in 2005, 2008, and 2015, respectively. He actively researched passive RFID systems and their new applications, ultra-low-power wireless communication systems, and efficient protocols at Korea University with global companies such as Samsung Electronics and LG Electronics. Since 2015, he has been the Principal Engineer with the Research and Development Group, RAONTECH, where he is responsible for the digital design of microdisplay systems. His current research interests include high definition microdisplay panel and controller design, low-power timing controller architecture of the panel, digital image processing, data compression, and high-speed interface for video streaming.

MIN-SEOK KIM received the B.S. and M.S. degrees from Korea University, in 1990 and 1992, respectively, and the Ph.D. degree in electrical engineering from Dankook University, in 2011. He was the Key Engineer with Samsung SDI and Iljin Display, developing display driving circuits, HTPS TFT-LCD, and LCoS display module for microdisplay applications. Since 2012, he has been the Head of the Research and Development Group, RAONTECH. He oversees the engineering departments and focuses on developing new products and technologies. He has 30 years of industry experience in micro display circuit design.

CHUL LEE (Member, IEEE) received the B.S., M.S., and Ph.D. degrees in electrical engineering from Korea University, Seoul, South Korea, in 2003, 2008, and 2013, respectively.

From 2002 to 2006, he was with Biospace Inc., Seoul, where he was involved in the development of medical equipment. From 2013 to 2014, he was a Postdoctoral Scholar at the Department of Electrical Engineering, Pennsylvania State University, University Park, PA, USA. From 2014 to 2015, he was a Research Scientist at the Department of Electrical and Electronic Engineering, The University of Hong Kong, Hong Kong. From 2015 to 2019, he was an Assistant Professor at the Department of Computer Engineering, Pukyong National University, Busan, South Korea. He joined the Department of Multimedia Engineering, Dongguk University, Seoul, in March 2019, where he is currently an Associate Professor. His current research interests include image processing and computational imaging with an emphasis on restoration and high dynamic range imaging.

Dr. Lee received the Best Paper Award from the *Journal of Visual Communication and Image Representation* in 2014. He is also an Editorial Board Member of the *Journal of Visual Communication and Image Representation*.

• • •