Head-Tracking for Virtual Reality Applications

Michael Anderson,* Jin Woo Park*
Stanford University, Stanford, California 94305

I. Introduction

Virtual reality is poised to change how the world consumes entertainment, communicates, and explores new worlds. Oculus VR wants consumers to experience time and space travel firsthand.[1] Magic Leap envisions a future of whales leaping out of gymnasium floors.[2] Vive promises immersive experiences in which the real world fades away.[3]

Many companies are working to bring virtual reality technology to the market. The industry has grown significantly in recent years, enabled by advances in low-cost sensors and high-resolution displays. The core challenge facing this expanding market is developing a seamless transition between the real and virtual world. For head-mounted displays, this requires accurately tracking the motion of the user's head. As the user looks about in the real world, the user's display of the virtual world must closely approximate their motion.

In this paper, we develop a head-tracking system for virtual reality applications. We consider different filtering systems, then implement an Extended Kalman Filter (EKF) with representative measurement models. We evaluate the accuracy of the system for various sensor placements and compensate accelerometer measurements for tangential and centripetal acceleration. We then incorporate a novel accelerometer constellation design into our system. This constellation might serve as a secondary source for angular velocity and acceleration measurements or replace drift-prone gyroscopes. Our results demonstrate the accelerometer constellation improves head-tracking performance, at the cost of additional computational complexity.

II. Review of Literature

Head-tracking systems must balance two fundamental needs: obtaining an accurate estimate of the user's head orientation, and minimizing the headset's latency interval. LaValle et al. define the latency interval to be the "the time from head movement to the appearance of the corresponding image on the retinas." The delay between taking sensor measurements at time t, estimating the headset orientation, and rendering an image to the user at $t + \delta t$ may degrade the virtual reality experience or induce motion sickness. LaValle and Abrash agree that a seamless virtual reality experience requires a latency interval of no more than 25ms.[10] For this reason, many researchers select computationally cheap filters that can quickly determine headset orientation to within a few degrees.[6]

We begin our study of different filter selections by considering commercially-available VR headsets. LaValle et al. outline the process by which the Oculus Rift headset tracks user head orientation and position.[6] The headset contains three-axis gyroscopes, accelerometers, and magnetometers that are each sampled at 1000Hz. Euler integration on sampled gyroscope measurements provides an initial head orientation estimate, but gyroscope drift introduces error into this dead-reckoned estimate over time; to reduce drift error, LaValle et al. fuse this orientation measurement with accelerometer data in a complementary filter. The advantage of this approach is that the estimator

*Masters Student, Department of Aeronautics and Astronautics

only must perform a few multiplications each update, but the filter's fixed gains must be carefully tuned.

Unlike the Oculus Rift team, LaViola forgoes complementary filters and considers the relative performance of Extended and Unscented Kalman Filters.[8] LaViola collects high-accuracy "ground truth" data of VR users rotating their heads and hands, then performs Monte Carlo analysis with simulated noisy tracking data. With a simulated sensor sample rate of 80Hz, the root mean square error (RMSE) for EKF and UKF is 0.297 and 0.304 degrees respectively. LaViola argues the performance of EKF and UKF are comparable for head-tracking applications, but UKF requires far more computations. In these simulations, EKF on average takes 266.13 ms/estimate, whereas UKF takes 3294.2 ms/estimate. Both results are well outside of the permissible latency interval discussed by LaValle and Abrash, but it is worth noting LaViola uses 4th order Runge-Kutta for all integration steps. LaViola ultimately recommends EKFs over UKFs for virtual reality head-tracking.

Motivated by LaViola's results, we focus our attention on EKF implementations for VR head-tracking. Sabatelli et al. implement a double-stage EKF using gyroscope, accelerometer, and magnetometer measurements.[9] In this system, the EKF's update step is separated into two parts: a pitch/roll update utilizing gyroscope and accelerometer measurements, and a yaw update based on magnetometer measurements. This separation of updates is intended to isolate the pitch and roll estimates from magnetic interference. However, there is some ambiguity in how Sabatelli combines the results of each update step. The proposed algorithm sets parts of each update quaternion equal to zero, then adds the two together to produce the *a posteriori* state estimate. This operation does not produce orientation quaternions of unit length, and it is also unclear if Sabatelli normalizes each quaternion before summing them. We experiment with this approach but do not present our findings here.

Many of the works discussed so far motivate their filtering systems with a discussion on gyroscope drift. While gyroscopes can provide reliable angular velocity measurements in the short term, these measurements may drift as a function of time and introduce errors into the orientation estimate. To remedy or mitigate this problem, we investigate using an array of accelerometers to extract angular velocity measurements.

Cardou et al. present an algorithm for estimating the angular velocity of a rigid body using an array of MEMS accelerometers.[14] The author proposes the Centripetal Acceleration Nonlinear Program (CANP) method to generate an angular acceleration matrix from an array of accelerometer readings. The symmetric and skew-symmetric parts of the matrix contain quadratic products of angular velocity and angular acceleration components respectively. The proposed algorithm minimizes the error in the angular acceleration matrix by evaluating the first order necessary conditions for optimality, as well as the matrix eigenvalues and eigenvectors. The angular velocity and acceleration of the rigid body can then be extracted. We implement this algorithm but use the approach discussed by Zou et al. in our head-tracking system.

Zou et al. also consider extracting angular velocity from accelerometer measurements.[15] The authors propose an isotropic-polyhedral layout of simplicial biaxial accelerometers (SBA), with a three-axis MEMS accelerometer placed at each face of the polyhedron. Zou et al. are able to decouple and extract relative tangential and centripetal accelerations due to the geometric isotropy of the layout and the structural planar isotropy of the SBA. Hence, the angular velocity of the rigid body could be estimated using the relative tangential acceleration information.

This research forms the basis on which we build our head-tracking system. LaValle et al. motivate our sensor selections and corresponding measurement models.[6] LaViola justifies the use of EKFs over UKFs for head orientation estimation and provides an EKF framework well-suited to this problem.[8] Sabatelli et al. describe the quaternion-based head rotation dynamics.[9] Finally, Cardou et al. and Zou et al. outline a framework for extracting angular velocity measurements

without the need for a gyroscope.[14,15]

III. System Kinematics

In defining the system kinematics, we restrict our attention to head orientation tracking; the user's head is free to rotate, but we assume the headset does not undergo any translation. This is a reasonable assumption in a virtual reality setting, in which users often control avatar translation via a controller.

We represent the headset orientation as a quaternion $q = (q_0, q_1, q_2, q_3)$ of unit length. The time rate of change of q can be expressed in terms of angular velocity $w = (w_x, w_y, w_z)$ by:

$$\dot{q} = Aq \ \text{, where } A = \frac{1}{2}\begin{bmatrix} 0 & -w_x & -w_y & -w_z \\ w_x & 0 & -w_z & w_y \\ w_y & w_z & 0 & -w_x \\ w_z & -w_y & w_x & 0 \end{bmatrix} \tag{1}$$

We discretize this equation using a first order Euler approximation for $\dot{q}$ to obtain the state update equation for our filter implementation:

$$q_{t+1} = (I_4 + A\delta t)q_t \tag{2}$$

The Jacobian of the state update equation can be expressed as:

$$J_q = \delta t \begin{bmatrix} \frac{1}{\delta t} & -\frac{1}{2}w_x & -\frac{1}{2}w_y & -\frac{1}{2}w_z \\ \frac{1}{2}w_x & \frac{1}{\delta t} & -\frac{1}{2}w_z & \frac{1}{2}w_y \\ \frac{1}{2}w_y & \frac{1}{2}w_z & \frac{1}{\delta t} & -\frac{1}{2}w_x \\ \frac{1}{2}w_z & -\frac{1}{2}w_y & \frac{1}{2}w_x & \frac{1}{\delta t} \end{bmatrix} \tag{3}$$

Note that the state evolution of the system is governed entirely by the quaternion kinematics, Equation 2. We do not make any assumptions about the system dynamics, such as modeling the system in terms of Euler's equations of rigid body rotation. The moments of inertia and input torques of our system are fundamentally unknown, and no dynamics models are provided to the filter.

IV. Sensors and Measurement Models

This section describes the parameters and measurement models for individual sensors. We restrict our attention to sensors common to many virtual reality devices: gyroscopes, magnetometers, and accelerometers.

A. Notation

The two frames of interest $\underrightarrow{\mathcal{F}}_a$ and $\underrightarrow{\mathcal{F}}_b$ are the inertial frame and headset body frame respectively. Furthermore, the superscripts $^{\bullet a}$ and $^{\bullet b}$ represent time derivatives w.r.t. the inertial and body frames, respectively. The angular velocity and acceleration of $\underrightarrow{\mathcal{F}}_b$ w.r.t. $\underrightarrow{\mathcal{F}}_a$ are denoted $\underrightarrow{\omega}^{ba}$ and $\underrightarrow{\dot{\omega}}^{ba}$ respectively. The position vector $\underrightarrow{r}^{jc}$ is the position of j relative to the kinematic center of the body c. Although we are not interested in $\underrightarrow{r}^{cw}$ and this quantity is impractical, $\underrightarrow{r}^{cw^{\bullet a \bullet a}}$ represents the translational acceleration of the body, meaning the second time derivative w.r.t. the inertial frame.

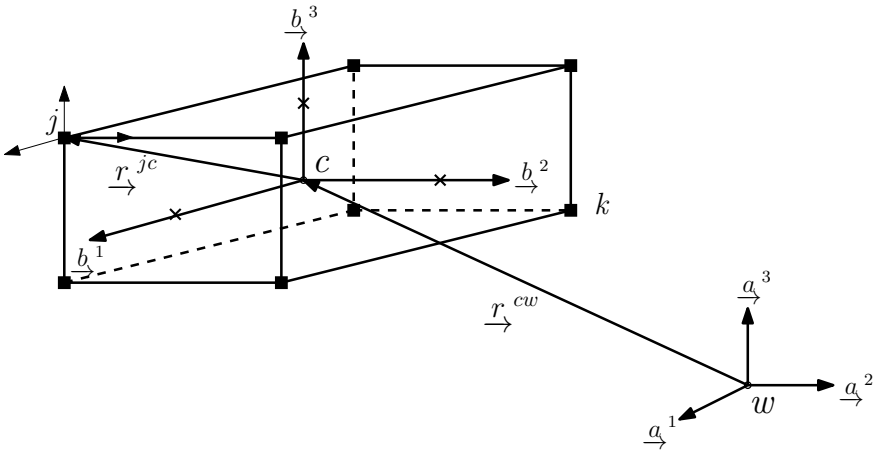

Figure 1. Overview of the Accelerometer Constellation.

B. Gyroscope Model

Simulated gyroscope measurements are generated according to the following model:

$$\hat{\omega}_b^g = \omega_b^{true} + \beta + v_t, \quad v_t \sim \mathcal{N}(0, R_g) \tag{4}$$

where ω_b^{true} is the true angular rotation rate in the headset body frame, β is the gyroscope static bias, and v_t models Gaussian white noise. In practice, the gyroscope static bias can be readily determined and removed from sensor measurements; we leave the full uncompensated static bias in Equation 4 in lieu of modeling the angle random walk.

C. Magnetometer Model

The magnetometer measurement model is given by:

$$\hat{m}_b = m_b^{true} + v_t, \quad v_t \sim \mathcal{N}(0, R_m) \tag{5}$$

where m_b^{true} is the true magnetic field vector in the headset body frame and v_t models Gaussian white noise.

D. Accelerometer Model

Multiple sources of acceleration affect accelerometer measurements, namely acceleration due to gravity as well as tangential and centripetal acceleration. Consider an accelerometer located at the corner j in Figure 1. The position of point j relative to point w is:

$$\underrightarrow{r}^{jw} = \underrightarrow{r}^{jc} + \underrightarrow{r}^{cw}.$$

The velocity of point j relative to point w w.r.t. $\underrightarrow{\mathcal{F}}_a$ is:

$$\underrightarrow{r}^{jw^{\bullet a}} = \left(\underrightarrow{r}^{jc^{\bullet b}} + \underrightarrow{\omega}^{ba} \times \underrightarrow{r}^{jc}\right) + \underrightarrow{r}^{cw^{\bullet a}}.$$

The acceleration of point j relative to point w w.r.t. $\underrightarrow{\mathcal{F}}_a$ is:

$$\begin{aligned}\underrightarrow{r}^{jw^{\bullet a \bullet a}} = &\left[\underrightarrow{r}^{jc^{\bullet b \bullet b}} + \left(\underrightarrow{\omega}^{ba^{\bullet b}} \times \underrightarrow{r}^{jc} + \underrightarrow{\omega}^{ba} \times \underrightarrow{r}^{jc^{\bullet b}}\right)\right] \\ &+ \left[\underrightarrow{\omega}^{ba} \times \underrightarrow{r}^{jc^{\bullet b}} + \underrightarrow{\omega}^{ba} \times \underrightarrow{\omega}^{ba} \times \underrightarrow{r}^{jc}\right] \\ &+ \underrightarrow{r}^{cw^{\bullet a \bullet a}}.\end{aligned}$$

This equation can be simplified for rigid bodies to be:

$$\underset{\rightarrow}{r}^{jw^{\bullet a \bullet a}} = \left(\underset{\rightarrow}{\omega}^{ba^{\bullet b}} + \underset{\rightarrow}{\omega}^{ba} \times \underset{\rightarrow}{\omega}^{ba} \right) \times \underset{\rightarrow}{r}^{jc} + \underset{\rightarrow}{r}^{cw^{\bullet a \bullet a}}. \tag{6}$$

The component of vectors expressed in $\underset{\rightarrow}{\mathcal{F}}_b$ are denoted with subscript b with bold text.

$$\underset{\rightarrow}{r}^{jw^{\bullet a \bullet a}} = \underset{\rightarrow}{\mathcal{F}}_b{}^{\mathsf{T}} \left[\left(\dot{\boldsymbol{\omega}}_b^{ba^\wedge} + \boldsymbol{\omega}_b^{ba^\wedge} \boldsymbol{\omega}_b^{ba^\wedge} \right) \mathbf{r}_b^{jc} + \ddot{\mathbf{r}}_b^{cw} \right],$$

$$\ddot{\mathbf{r}}_b^{jw} = \mathbf{W}_b \mathbf{r}_b^{jc} + \ddot{\mathbf{r}}_b^{cw},$$

where

$$\mathbf{W}_b = \dot{\boldsymbol{\omega}}_b^{ba^\wedge} + \boldsymbol{\omega}_b^{ba^\wedge} \boldsymbol{\omega}_b^{ba^\wedge} \tag{7}$$

is the angular acceleration matrix resolved in body frame $\underset{\rightarrow}{\mathcal{F}}_b$. Therefore, the accelerometer readouts are $\ddot{\mathbf{r}}_b^{jw}$ for $j = 1, \ldots, n$ and are resolved in $\underset{\rightarrow}{\mathcal{F}}_b$.

Hence, our accelerometer model is

$$\hat{\mathbf{r}}_b^{jw^{\bullet a \bullet a}} = \mathbf{r}_b^{cw^{\bullet a \bullet a}} + \left(\dot{\boldsymbol{\omega}}_b^{ba^\wedge} \, \mathbf{r}_b^{jc} \right) + \boldsymbol{\omega}_b^{ba^\wedge} \left(\boldsymbol{\omega}_b^{ba^\wedge} \, \mathbf{r}_b^{jc} \right) + v_t, \quad v_t \sim \mathcal{N}(0, R_a) \tag{8}$$

where $\mathbf{r}_b^{cw^{\bullet a \bullet a}}$ is the true gravity vector in the headset body frame (no translational motion), $\boldsymbol{\omega}_b^{ba}$ is the angular velocity of the headset, $\mathbf{r}_b^{jc}$ describes the offset of the sensor from the center of rotation, and v_t models Gaussian white noise. Note that if the sensor is positioned at the center of rotation, $\mathbf{r}_b^{jc} = (0, 0, 0)$ and the noisy acceleration due to gravity can be directly measured.

E. Sensor Noise Parameters

Realistic noise profiles for each sensor type were determined by collecting data from a stationary MPU-6050 MEMS IMU and HMC5883L MEMS magnetometer over a 30 minute period. The goal of this short data collection period was to obtain the Gaussian noise parameters for each sensor and approximate gyroscope static bias; a longer data collection period would be required to fully characterize the angle random walk, bias instability, and rate random walk for the gyroscope measurement model. The static bias of the gyroscope model and Gaussian white noise covariance matrices for each sensor type are given below:

$$\beta = (-0.0388, 0.1440, -0.1383) \ \text{deg/s}$$

$$R_g = \begin{bmatrix} 0.0073 & 0 & 0 \\ 0 & 0.0091 & 0 \\ 0 & 0 & 0.0075 \end{bmatrix}, \quad R_m = 10^{-6} \begin{bmatrix} 0.4017 & -0.0353 & 0.0255 \\ -0.0353 & 0.3963 & -0.0061 \\ 0.0255 & -0.0061 & 0.3319 \end{bmatrix}$$

$$R_a = 10^{-4} \begin{bmatrix} 0.1042 & 0 & 0.0004 \\ 0 & 0.1010 & -0.0005 \\ 0.0004 & -0.0005 & 0.2463 \end{bmatrix}$$

V. Filter Design

We implement an Extended Kalman Filter as part of our head-tracking system. The state variable q represents the headset orientation at each time step. The true gravity and magnetic field vectors in the headset body frame are obtained by applying q to the corresponding vectors in the inertial frame, i.e. multiplying by R_a^b.

The gravity vector in the inertial frame is assumed to point directly downward, and we use the assumption described in Sabatelli et al.[9] that the magnetic field vector in the inertial frame is directed along the y-axis only. These assumptions simplify the resulting Jacobians, but could readily be relaxed by using a gravitational and geomagnetic model.

$$\hat{g}_b = R_a^b \begin{bmatrix} 0 \\ 0 \\ |g| \end{bmatrix} = \begin{bmatrix} q_0^2+q_1^2-q_2^2-q_3^2 & 2q_1q_2+2q_0q_3 & 2q_1q_3-2q_0q_2 \\ 2q_1q_2-2q_0q_3 & q_0^2-q_1^2+q_2^2-q_3^2 & 2q_2q_3+2q_0q_1 \\ 2q_1q_3+2q_0q_2 & 2q_2q_3-2q_0q_1 & q_0^2-q_1^2-q_2^2+q_3^2 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ |g| \end{bmatrix} = |g| \begin{bmatrix} 2q_1q_3-2q_0q_2 \\ 2q_0q_1+2q_2q_3 \\ q_0^2-q_1^2-q_2^2+q_3^2 \end{bmatrix} \tag{9}$$

$$\hat{m}_b = R_a^b \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 2q_1q_2+2q_0q_3 \\ q_0^2-q_1^2+q_2^2-q_3^2 \\ 2q_2q_3-2q_0q_1 \end{bmatrix} \tag{10}$$

We take the derivative of $\hat{g}^b$ and $\hat{m}^b$ with respect to the quaternion components to obtain the corresponding Jacobian matrices:

$$J_a = |g| \begin{bmatrix} -2q_2 & 2q_3 & -2q_0 & 2q_1 \\ 2q_1 & 2q_0 & 2q_3 & 2q_2 \\ 2q_0 & -2q_1 & -2q_2 & 2q_3 \end{bmatrix}, \quad J_m = \begin{bmatrix} 2q_3 & 2q_2 & 2q_1 & 2q_0 \\ 2q_0 & -2q_1 & 2q_2 & -2q_3 \\ -2q_1 & -2q_0 & 2q_3 & 2q_2 \end{bmatrix}$$

All together, the measurement update and Jacobian matrix can be written as:

$$y = \begin{bmatrix} \hat{g}_b \\ \hat{m}_b \end{bmatrix}, \quad J = \begin{bmatrix} J_a & 0_{3x4} \\ 0_{3x4} & J_m \end{bmatrix}$$

We employ the EKF algorithm outlined by Thrun et al.[5] The filter prediction step is carried out by applying the angular rate measurement received from the gyroscope or accelerometer constellation to the current state estimate. The expected accelerometer and magnetometer measurements at the estimated orientation are then computed using Equations 9-10. The block diagonal Jacobian matrix J is then evaluated at the estimated orientation and used as part of the filter update step.

We consider the performance of our system with four different sensor configurations, explained below:

1. Center of Head Measurements: Gyroscope, Magnetometer, Accelerometer

The Center of Head configuration represents the "ideal" sensor placement, in which the accelerometer is mounted at the center of rotation of the user's head. This is extremely impractical for a head-mounted VR system, but it is a useful test case. At the center of rotation, $\mathbf{r}_b^{jc} = (0,0,0)$ and the accelerometer measurement equation simplifies to:

$$\hat{\mathbf{r}}_b^{jw^{\bullet a \bullet a}} = \mathbf{r}_b^{cw^{\bullet a \bullet a}} + v_t, \quad v_t \sim \mathcal{N}(0, R_a) \tag{11}$$

This provides a noisy measurement of the gravity vector and directly conveys headset orientation information. In the subsequent configurations, additional work must be performed to isolate the gravity vector.

The angular velocity at any point on a rotating rigid body is the same, and so the placement of the gyroscope is unimportant. Similarly, we assume the Earth's magnetic field is locally uniform and placement of the magnetometer does not affect sensor readings. The axes of all sensors are assumed to be aligned with the headset body frame.

2. "Naive" Head-Mounted Measurements: Gyroscope, Magnetometer Accelerometer

The "Naive" Head-Mounted configuration offsets the headset sensor package 4cm in each axis from the center of rotation, $\mathbf{r}_b^{jc} = (0.04, 0.04, -0.04)$. With the accelerometer offset from the center of rotation, the sensor will experience acceleration due to gravity as well as acceleration due to rotation. In the naive configuration, the system makes no attempt to compensate for the additional terms in the full accelerometer model. The filter assumes the received accelerometer measurement is a noisy gravity vector and proceeds as before.

3. "Compensated" Head-Mounted Measurements: Gyroscope, Magnetometer Accelerometer

In the "Compensated" Head-Mounted configuration, the system attempts to compensate for the additional terms in the accelerometer model before passing the measurement to the filter. We estimate the terms $\left(\dot{\boldsymbol{\omega}}_b^{ba\wedge}\,\mathbf{r}_b^{jc}\right)$ and $\boldsymbol{\omega}_b^{ba\wedge}\left(\boldsymbol{\omega}_b^{ba\wedge}\,\mathbf{r}_b^{jc}\right)$ using gyroscope measurements, where $\dot{\boldsymbol{\omega}}_b^{ba\wedge}$ is calculated with a first order Euler approximation for the derivative. In future work, we would like to use a higher order derivative approximation.

Once the angular velocity and angular acceleration-dependent terms have been estimated, we subtract these values from the measured acceleration and pass the resulting gravity vector approximation to the filter.

4. Constellation Head-Mounted Measurements: Accelerometer Constellation, Magnetometer

In the Constellation Head-Mounted configuration, the gyroscope has been removed from the system. Angular rate measurements are determined by the constellation of accelerometers, mounted at various points around the user's head. We explain the design of the constellation next.

VI. Accelerometer Constellation

Head-tracking systems often rely on angular velocity measurements, but gyroscopes are particularly susceptible to drift. This can adversely impact the system and degrade the quality of the estimated head orientation. In contrast, accelerometers are less susceptible to drift. Furthermore, so-called simplicial triaxial accelerometers (STA) are less prone to bias than gyroscopes, which is why various researchers have investigated the use of STA strapdowns and linear accelerometers for angular velocity computations by exploiting geometric and kinematics relationships. We estimate angular velocity measurements from an array of accelerometers using the trapezoidal integration method discussed by Zou et al.[15]

Figure 1 depicts the constellation of accelerometers. The point c on the rigid body is the kinematic center. A total of eight STAs are placed on each corner of the rectangular prism to measure acceleration in three orthogonal directions. The following sections extract the angular velocity and acceleration of a rigid body given accelerometer sensor measurements $\underrightarrow{r}^{jw^{\bullet a\bullet a}}$, $j = 1, \ldots, 8$.

A. Translational Acceleration

The centroid (i.e., kinematic center) acceleration (henceforth translational acceleration) of the body $\underrightarrow{r}^{cw^{\bullet a \bullet a}}$ can be calculated by summing the acceleration of two points that are symmetric about all three body axes $\underrightarrow{\mathcal{F}}_b$. Note that $\underrightarrow{r}^{jc} = -\underrightarrow{r}^{kc}$ in Fig. 1. Hence, summing the acceleration of points j and k with respect to inertial point resolved in the body frame, i.e. $\ddot{\mathbf{r}}_b^{jw}$ and $\ddot{\mathbf{r}}_b^{kw}$ respectively, results in $2\,\ddot{\mathbf{r}}_b^{cw}$:

$$\begin{aligned}
\underrightarrow{r}^{jw^{\bullet a \bullet a}} + \underrightarrow{r}^{kw^{\bullet a \bullet a}} &= \underrightarrow{\mathcal{F}}_b{}^{\mathsf{T}} \left(\ddot{\mathbf{r}}_b^{jw} + \ddot{\mathbf{r}}_b^{kw} \right), \\
\underrightarrow{\mathcal{F}}_b{}^{\mathsf{T}} \left(\ddot{\mathbf{r}}_b^{jw} + \ddot{\mathbf{r}}_b^{kw} \right) &= \underrightarrow{\mathcal{F}}_b{}^{\mathsf{T}} \left[\mathbf{W}_b \cancelto{\mathbf{0}_b}{\left(\mathbf{r}_b^{jc} + \mathbf{r}_b^{kc} \right)} + 2\,\ddot{\mathbf{r}}_b^{cw} \right], \\
\underrightarrow{r}^{jw^{\bullet a \bullet a}} + \underrightarrow{r}^{kw^{\bullet a \bullet a}} &= 2\,\underrightarrow{\mathcal{F}}_b{}^{\mathsf{T}} \ddot{\mathbf{r}}_b^{cw}.
\end{aligned} \tag{12}$$

Therefore, the translational acceleration vector $\underrightarrow{r}^{cw^{\bullet a \bullet a}}$ can be obtained as shown below, where n denotes the number of STAs.

$$\underrightarrow{r}^{cw^{\bullet a \bullet a}} = \frac{1}{n} \sum_{i=1}^{n} \underrightarrow{r}^{iw^{\bullet a \bullet a}}. \tag{13}$$

B. Angular Acceleration Matrix

The angular acceleration matrix $\mathbf{W}_b$ is the matrix representing the tangential and centripetal acceleration of a rigid body resolved in body frame $\underrightarrow{\mathcal{F}}_b$. We denote the tangential and centripetal component of the angular acceleration matrix as $\mathbf{W}_b^{\parallel}$ and $\mathbf{W}_b^{\perp}$ respectively. The rotational motion of the rigid body relative to the kinematic center can be represented as:

$$\begin{aligned}
\underrightarrow{r}^{jc^{\bullet a \bullet a}} &= \underrightarrow{r}^{jw^{\bullet a \bullet a}} - \underrightarrow{r}^{cw^{\bullet a \bullet a}}, \\
\underrightarrow{\mathcal{F}}_b{}^{\mathsf{T}} \ddot{\mathbf{r}}_b^{jc} &= \underrightarrow{\mathcal{F}}_b{}^{\mathsf{T}} \mathbf{W}_b \mathbf{r}_b^{jc}, \\
\ddot{\mathbf{r}}_b^{jc} &= \left(\dot{\boldsymbol{\omega}}_b^{ba^{\times}} + \boldsymbol{\omega}_b^{ba^{\times}} \boldsymbol{\omega}_b^{ba^{\times}} \right) \mathbf{r}_b^{jc}, \\
\ddot{\mathbf{r}}_b^{jc} &= \left(\mathbf{W}_b^{\parallel} + \mathbf{W}_b^{\perp} \right) \mathbf{r}_b^{jc}.
\end{aligned} \tag{14}$$

The angular acceleration matrix is equivalent to:

$$\begin{aligned}
\mathbf{W}_b = &\begin{bmatrix} 0 & -\dot{\omega}_3 & \dot{\omega}_2 \\ \dot{\omega}_3 & 0 & -\dot{\omega}_1 \\ -\dot{\omega}_2 & \dot{\omega}_1 & 0 \end{bmatrix} \\
&+ \begin{bmatrix} -{\omega_2}^2 - {\omega_3}^2 & \omega_1\omega_2 & \omega_1\omega_3 \\ \omega_2\omega_1 & -{\omega_1}^2 - {\omega_3}^2 & \omega_2\omega_3 \\ \omega_3\omega_1 & \omega_3\omega_2 & -{\omega_1}^2 - {\omega_2}^2 \end{bmatrix}.
\end{aligned} \tag{15}$$

C. Extracting the Angular Acceleration Matrix

According to Eq. (14), deriving $\mathbf{W}_b$ seems straightforward. Nonetheless, $\mathbf{r}_b^{jc}$ is not invertible and the Moore-Penrose inverse of $\mathbf{r}_b^{jc}$ will result in a singularity. A congregated matrix of position and

acceleration vectors were given by Zou et al.[15] Hence, two congregated $3 \times n$ vectrices $\mathbf{\Pi}_b$ and $\ddot{\mathbf{\Pi}}_b$ resolved in the body frame $\underrightarrow{\mathcal{F}}_b$, are introduced:

$$\mathbf{\Pi}_b = \begin{bmatrix} r_{b1}^{1c} & \cdots & r_{b1}^{nc} \\ r_{b2}^{1c} & \cdots & r_{b2}^{nc} \\ r_{b3}^{1c} & \cdots & r_{b3}^{nc} \end{bmatrix}, \quad \ddot{\mathbf{\Pi}}_b = \begin{bmatrix} \ddot{r}_{b1}^{1c} & \cdots & \ddot{r}_{b1}^{nc} \\ \ddot{r}_{b2}^{1c} & \cdots & \ddot{r}_{b2}^{nc} \\ \ddot{r}_{b3}^{1c} & \cdots & \ddot{r}_{b3}^{nc} \end{bmatrix}.$$

Equation (14) can be re-written with $\mathbf{\Pi}_b$ and $\ddot{\mathbf{\Pi}}_b$ as follows:

$$\ddot{\mathbf{\Pi}}_b^{\mathsf{T}} = \mathbf{W}_b \mathbf{\Pi}_b.$$

The angular acceleration matrix can be extracted by post-multiplying $\mathbf{\Pi}_b^{\mathsf{T}}$ on both sides:

$$\ddot{\mathbf{\Pi}}_b \mathbf{\Pi}_b^{\mathsf{T}} = \mathbf{W}_b \mathbf{\Pi}_b \mathbf{\Pi}_b^{\mathsf{T}}.$$

Consequently, the angular acceleration matrix can be evaluated below:

$$\mathbf{W}_b = \ddot{\mathbf{\Pi}}_b \mathbf{\Pi}_b^{\mathsf{T}} \left(\mathbf{\Pi}_b \mathbf{\Pi}_b^{\mathsf{T}} \right)^{-1}. \tag{16}$$

The resulted angular acceleration matrix $\mathbf{W}_b$ is a 3×3 matrix.

The tangential $\mathbf{W}_b^{\parallel}$ and centripetal $\mathbf{W}_b^{\perp}$ component of the angular acceleration matrix are

$$\mathbf{W}_b^{\perp} = \frac{1}{2} \left(\mathbf{W}_b + \mathbf{W}_b^{\mathsf{T}} \right), \tag{17}$$

and

$$\mathbf{W}_b^{\parallel} = \mathbf{W}_b - \frac{1}{2} \left(\mathbf{W}_b + \mathbf{W}_b^{\mathsf{T}} \right) = \mathbf{W}_b - \mathbf{W}_b^{\perp}. \tag{18}$$

D. Tangential Acceleration Method

This method uses a time integration of the tangential component $\mathbf{W}_b^{\parallel}$ of the angular acceleration matrix $\mathbf{W}_b$ (15). Since $\mathbf{W}_b^{\parallel}$ is a skew symmetric matrix, its axial vector (i.e., uncross) $\dot{\boldsymbol{\omega}}_b^{ba}$ needs to be evaluated for the integration.

$$\dot{\boldsymbol{\omega}}_b^{ba} = \mathbf{W}_b^{\parallel \vee} = \frac{1}{2} \begin{bmatrix} \mathbf{W}_{b_{3,2}}^{\parallel} - \mathbf{W}_{b_{2,3}}^{\parallel} \\ \mathbf{W}_{b_{1,3}}^{\parallel} - \mathbf{W}_{b_{3,1}}^{\parallel} \\ \mathbf{W}_{b_{2,1}}^{\parallel} - \mathbf{W}_{b_{1,2}}^{\parallel} \end{bmatrix}.$$

The angular velocity can be summarized symbolically as:

$$\hat{\boldsymbol{\omega}}_b^{ba}(t) = \hat{\boldsymbol{\omega}}_b^{ba}(0) + \int_0^t \hat{\dot{\boldsymbol{\omega}}}_b^{ba}(\tau)\, d\tau,$$

where $\hat{(\cdot)}$ is an estimate of $(\cdot)$. We use the trapezoidal rule to perform this integration, inspired by Cardou et al. and Zou et al.[14,15]

VII. Results

A. Constellation Validation

The following plots show the performance of the constellation algorithm with only rotation about the y-axis. The results indicate that the algorithm estimates the angular velocity fairly well, though some drift accumulates from noise integration. The translational acceleration in the body frame correctly depicts the gravity vector in the inertial frame and is corroborated in Figure 4.

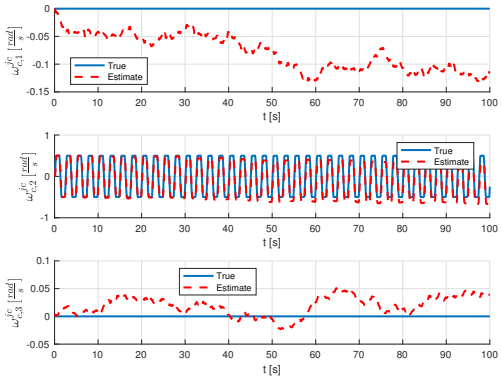

Figure 2. Estimated rigid body angular velocity.

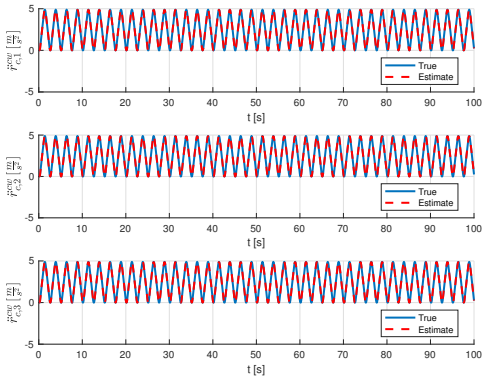

Figure 3. Translational acceleration of rigid body in the body frame.

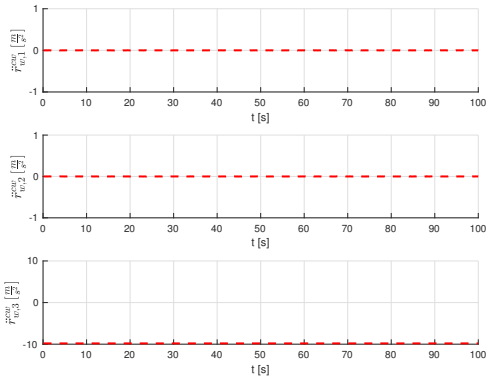

Figure 4. Translational acceleration of rigid body in the inertial frame.

B. Filter Performance

The following filter results are shown with the headset rotating in all three axes. The simulation framework supports any input angular velocity signal; for these simulations, the angular velocity of all three axes was chosen to be a sine wave with a 144 deg/s amplitude and 2.5s period (parameters determined by the authors rotating their heads). The initial state estimate is initialized to a random quaternion of unit length.

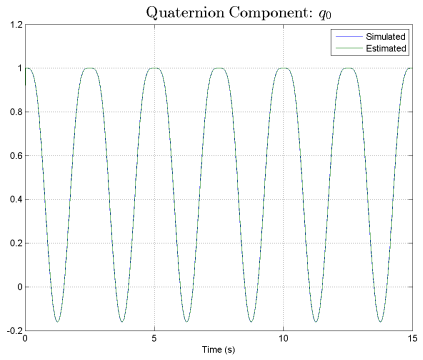

Figure 5. Constellation simulation: q_0

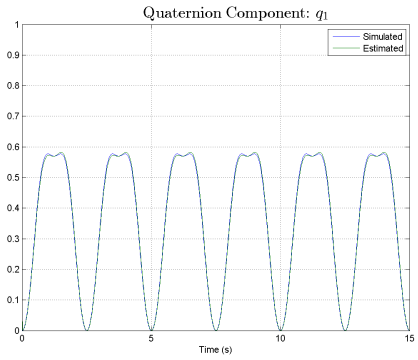

Figure 6. Constellation simulation: q_1

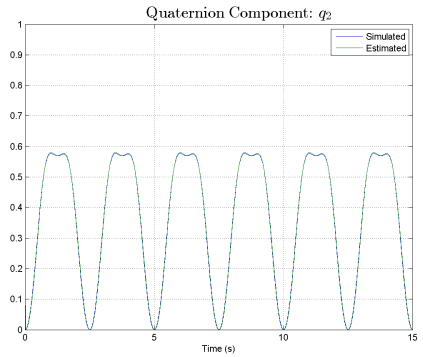

Figure 7. Constellation simulation: q_2

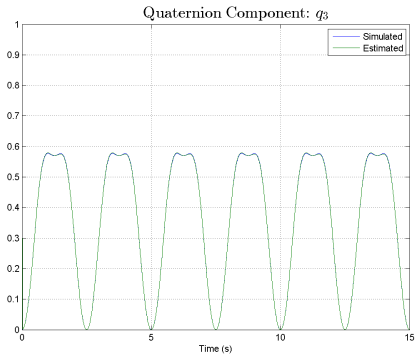

Figure 8. Constellation simulation: q_3

The head-tracking system quickly converges to the true head orientation in all of our simulations, with each of the four system configurations. However, the "compensated" system does experience some jitter in the state estimate, likely due to our first order Euler approximation for $\dot{\boldsymbol{\omega}}_b^{ba\wedge}$. In noiseless simulations, the compensated system is on par with the accelerometer constellation. In some simulations with noise, the compensated system performs worse than the naive implementation.

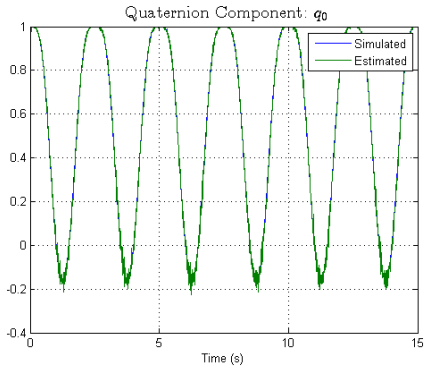

Figure 9. Compensated simulation: q_0

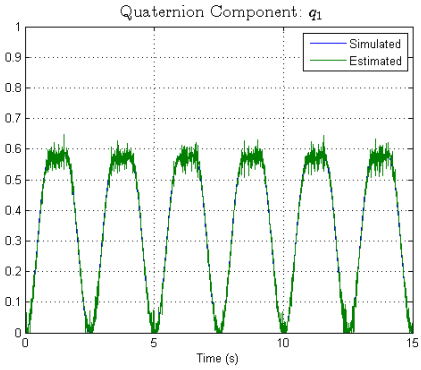

Figure 10. Compensated simulation: q_1

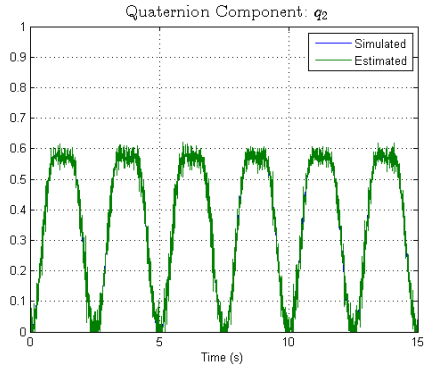

Figure 11. Compensated simulation: q_2

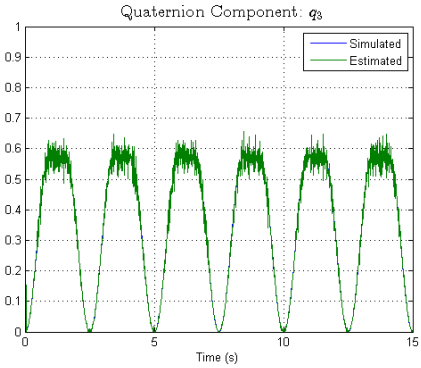

Figure 12. Compensated simulation: q_3

We look at the error between the simulated and estimated Euler angles to visualize the performance of each configuration:

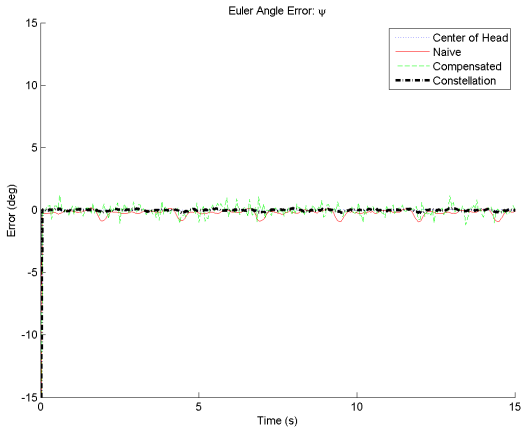

Figure 13. Simulated vs. Estimated: Yaw Error

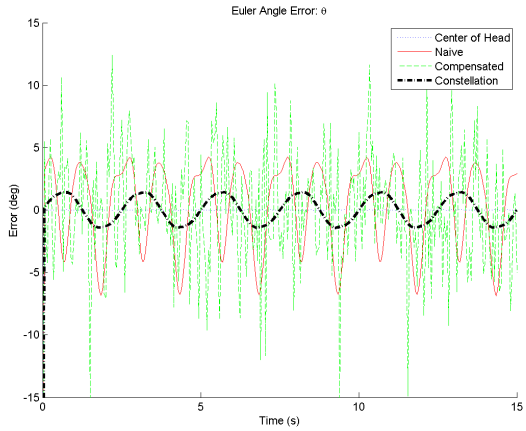

Figure 14. Simulated vs. Estimated: Pitch Error

Of the four configurations, the "ideal" configuration with Center of Head measurements per-

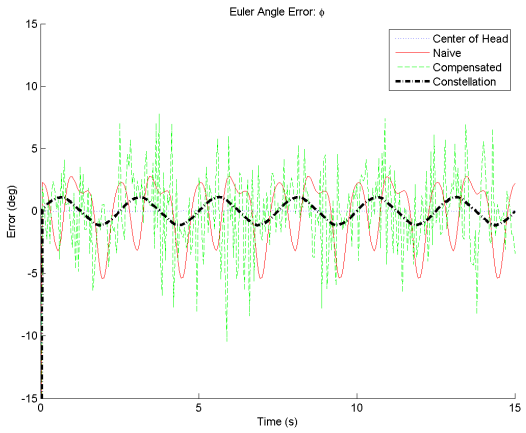

Figure 15. Simulated vs. Estimated: Roll Error

forms best. This matches our expectations, since the noisy gravity vector can be directly measured in this configuration. The accelerometer constellation configuration also tracks the head orientation well. After the initial estimate converges to the true orientation, the constellation system tracks the headset orientation to within 1.44 degrees in all axes. This level of accuracy is well within acceptable bounds for commercial virtual reality systems.

In comparison, the naive and compensated configurations struggle to track the true headset orientation. We expect the naive system to perform poorly, as the filter is always being fed an uncompensated measurement of the gravity vector. The compensated state estimate is also very noisy, likely due to our numerical scheme for $\dot{\boldsymbol{\omega}}_b^{ba^\wedge}$.

The yaw error shown in Figure 13 is especially noteworthy. Many headsets rely solely on magnetometers to estimate yaw because the gravity vector does not provide any yaw information.[6] In all four configurations, we see the yaw error is nearly 0 degrees. There are small fluctuations due to noise, but as expected the accelerometer model in each configuration does not dramatically impact the yaw estimate.

The root mean square error (RMSE) and filter computation times provide useful metrics for comparing each configuration. The constellation system requires more than twice the computation time than the naive and compensated configurations, but each system is well within the 20ms latency period discussed in the literature.

	ψ RMSE (deg)	θ RMSE (deg)	ϕ RMSE (deg)	Single Iter Comp (ms)
Center of Head	0.0354	0.0236	0.0173	0.2107
Naive	0.3097	3.3644	2.4064	0.2200
Compensated	0.3910	4.9697	2.9544	0.3311
Constellation	0.0752	1.0250	0.7659	0.8950

VIII. Conclusion

The four configurations considered in this paper all show promising attributes for virtual reality head-tracking applications. The performance of the accelerometer constellation is particularly exciting; while this configuration requires more than twice the computation time of other methods, the entire system is still fast enough to be integrated into a real VR rendering pipeline.

There are several directions for future work on this system. One opportunity is to fuse the

angular rate measurements from a gyroscope with the accelerometer constellation. Another interesting study would be to implement a predictive scheme to compensate for the difference between the sensor measurement time and filter update time. Finally, we would like to deploy these systems on hardware!

References

[1] "Oculus Rift." *Oculus VR*. Oculus Rift, LLC. Web. Accessed 6 May 2017. https://www.oculus.com/rift/

[2] "About Us." *Magic Leap*. Magic Leap, Inc. Web. Accessed 6 May 2017. https://www.magicleap.com/#/company

[3] "Discover Virtual Reality Beyond Imagination." *Vive*. HTC Corporation. Web. Accessed 6 May 2017. https://www.vive.com/us/

[4] Ercan, A.O., and Erdem, A.T., "On Sensor Fusion for Head Tracking in Augmented Reality Applications," *American Control Conference*, June 2011, pp. 1286-1291, http://ieeexplore.ieee.org/abstract/document/5991077/

[5] Thrun, S., Burgard, W., and Fox, D., *Probabilistic Robotics*, MIT Press, 2006.

[6] LaValle, S.M., Yershova, A., Katsev, M., and Antonov, M., "Head tracking for the Oculus Rift," *IEEE International Conference on Robotics and Automation (ICRA)*, May 2014, pp. 187-194, http://msl.cs.illinois.edu/~lavalle/papers/LavYerKatAnt14.pdf

[7] Zhang, Z.Q., Meng, X.L., and Wu, J.K., "Quaternion-Based Kalman Filter with Vector Selection for Accurate Orientation Tracking," *IEEE Transactions on Instrumentation and Measurement*, Vol. 61, No. 10, October 2012, pp. 2817-2824. doi: 10.1109/TIM.2012.2196397, http://ieeexplore.ieee.org/document/6213549/

[8] LaViola, J.J., "A Comparison of Unscented and Extended Kalman Filtering for Estimating Quaternion Motion," *Proceedings of the American Control Conference*, June 2003. doi: 10.1109/ACC.2003.1243440, http://ieeexplore.ieee.org/abstract/document/1243440/

[9] Sabatelli, S., Galgani, M., Fanucci, L., and Rocchi, A., "A Double-Stage Kalman Filter for Orientation Tracking With an Integrated Processor in 9-D IMU," *IEEE Transactions on Instrumentation and Measurement*, Vol. 62, No. 3, March 2013, pp. 590-598. doi: 10.1109/TIM.2012.2218692, http://ieeexplore.ieee.org/abstract/document/6316172/

[10] Abrash, M. "What VR Could, Should, and Almost Certainly Will Be Within Two Years," *Steam*. Valve, 2014. Web. Accessed 11 Jun 2017. http://media.steampowered.com/apps/abrashblog/Abrash%20Dev%20Days%202014.pdf

[11] Larsen, T.D., Andersen, N.A., Ravn, O., and Poulsen, N.K, "Incorporation of time delayed measurements in a discrete-time Kalman filter," *Proceedings of the 37th IEEE Conference on Decision & Control*, December 1998, Tampa, FL, http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=761918

[12] Lu, X., Xie, L., Zhang, H., and Wang, W., "Robust Kalman Filtering for Discrete-Time Systems With Measurement Delay," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 54, June 2007, pp. 522-526, http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=4237369

[13] Luinge, H. J. and Veltink, P. H., "Measuring orientation of human body segments using miniature gyroscopes and accelerometers," *Medical and Biological Engineering and Computing*, vol 43. April 2005, pp. 273-282, http://link.springer.com/content/pdf/10.1007%2FBF02345966.pdf

[14] Cardou, P., Fournier, G., and Gagnon, P., "Estimating the angular velocity of a rigid body moving in the plane from tangential and centripetal acceleration measurements," *IEEE/ASME Transactions on Mechatronics*, vol. 5, October 2011, pp. 1-13, http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=5551205

[15] Zou, T. and Angeles J., "Isotropic Accelerometer Strapdowns and Related Algorithms for Rigid-Body Pose and Twist Estimation," *Journal of Applied Mechanics*, vol. 81, November 2014, pp. 1-13, http://appliedmechanics.asmedigitalcollection.asme.org/data/journals/jamcav/930885/jam_081_11_111003.pdf