



# Orientation Correction for a 3D Hand Motion Tracking Interface Using Inertial Measurement Units

Nonnarit O-larnnithipong<sup>(✉)</sup>, Armando Barreto,  
Sudarat Tangnimitchok, and Neeranut Ratchatanantakit

Electrical and Computer Engineering Department,  
Florida International University, Miami, FL 33174, USA  
{nolar002, barretoa, stang018, nratc001}@fiu.edu

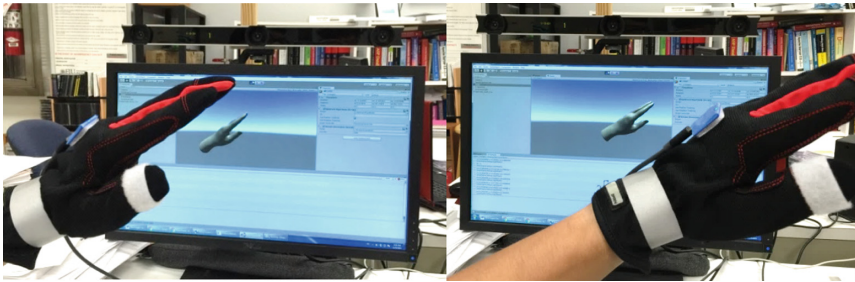
**Abstract.** This paper outlines the use of an orientation correction algorithm for a miniature commercial-grade Inertial Measurement Unit to improve orientation tracking of human hand motion and also to improve 3D User Interfaces experience to become more realistic. The algorithm uses the combination of gyroscope, accelerometer and magnetometer measurements to eliminate the drift in orientation measurement which is caused by the accumulation of the bias offset error in the gyroscope readings. The algorithm consists of three parts, which are: (1) bias offset estimation, (2) quaternion correction using gravity vector and magnetic North vector, and (3) quaternion interpolation. The bias offset estimation is performed during periods when the sensor is estimated to be static, when the gyroscope reading would provide only the bias offset error for prediction. The quaternion was calculated based on unbiased angular velocity and then used to rotate the gravity vector and magnetic North vector in the Earth's frame resulting in the calculated gravity vector and magnetic North vector in the sensor's frame. The angular errors between calculated and measured gravity vector and the angle between calculated and measured magnetic North vector are used to calculate the correction quaternion that must be applied to the previous quaternion result. The result of the orientation estimation using this algorithm can be used to track the orientation of human hand motion with less drift and improved orientation accuracy than achieved with the on-board Kalman-based orientation filtering.

**Keywords:** Inertial measurement unit · Gyroscope drift  
Orientation correction · Bias offset error estimation  
Quaternion correction using gravity vector · 3D hand motion tracking interface

## 1 Introduction

Most of the applications for 3D User Interfaces in Augmented Reality (AR) and Virtual Reality (VR) will require the user to interact with immersive environments. One of the key goals of VR and AR systems is to foster the perception of “presence” in the user, which strives to provide the user with highly realistic visual and acoustic output [1–4].

In some VR environments, the perception of presence may be severely disrupted when the user is asked to interact using highly “unnatural” sequences of actions performed on an input device (e.g., joystick, gamepad, mouse or keyboard). Thus, a full hand motion tracking system would significantly help in circumventing these current limitations and improve 3D User Interfaces to become more realistic. Our proposed hand motion tracking interface seeks to determine the movement of the human hand by combining two different sources of information: orientation tracking using Inertial Measurement Units (IMUs) and position tracking using infrared cameras: OptiTrack V120 Trio as shown in Fig. 1.



**Fig. 1.** Position tracking of hand motion visualized in unity

Our research has yielded an algorithm to minimize the gyroscope drift problem which commonly occurs in miniature commercial-grade IMUs. IMUs contain gyroscopes, which provide angular velocity measurements that can be used to track orientation of the object that they are attached to. However, orientation tracking also requires the mathematical integration of angular velocity readings from the gyroscopes. Most low-cost gyroscopes may output an erroneous non-zero measurement level even when they are not moving, called the bias offset error. This bias offset error can produce an orientation tracking error called drift, which causes many problems in navigation and other applications that use low-cost gyroscopes to determine the orientation of moving parts [5, 6]. Several studies and algorithms have been proposed to eliminate gyroscope drift error in inertial measurement systems. Most of the studies use Kalman-based processes to solve the error in inertial sensors [7–9]. But this approach is complex and can be complicated to implement [10, 11].

To estimate the orientation, many studies use the idea of sensor fusion which is the method that combines more than one sensor to determine one measurement. For example, a sensor fusion approach may suggest using gyroscopes, accelerometers and magnetometers to determine orientation [12, 13]. The idea of sensor fusion is also the main method to determine the orientation in this research. This paper outlines the use of this algorithm to improve orientation tracking of human hand motion by using the combination of gyroscope, accelerometer and magnetometer measurements.

## 2 Methodology

### 2.1 Bias Offset Estimation

To minimize the drift, an algorithm to correct the gyroscope bias offset has been proposed. With the idea that the IMU sensor should provide the reading of zero when there is no input or movement taking place, the bias offset error could be determined and used to subtract from future gyroscope readings, to obtain unbiased gyroscope data as a result. The predicted bias offset error ( $\hat{b}$ ) of each gyroscope axis is determined during the sensor's static periods by using a simple linear regression model [14] as shown in Eq. (1). The model coefficients ( $\beta_1$  and  $\beta_0$ ) can be found by using Eqs. (2) and (3), where  $b_i$  and  $t_i$  are the measured gyroscope bias and time at the  $i^{\text{th}}$  sample, respectively.

$$\hat{b} = \beta_0 + \beta_1 t \quad (1)$$

$$\beta_1 = \frac{\sum_{i=1}^n (t_i - \bar{t})(b_i - \bar{b})}{\sum_{i=1}^n (t_i - \bar{t})^2} \quad (2)$$

$$\beta_0 = \bar{b} - \beta_1 \bar{t} \quad (3)$$

The predicted bias is subtracted from the measured gyroscope reading, yielding the unbiased angular rate ( $\omega_B$ ). To avoid gimbal lock problems, the orientation representation used in this project is quaternion notation, placing the real part in the fourth component. The quaternion rate ( $\dot{q}$ ) can be calculated by using the unbiased angular rate ( $\omega_B$ ), where  $\hat{q}_0$  is the previous quaternion estimation of the orientation (For the first iteration of an algorithm,  $\hat{q}_0$  is set to  $[0, 0, 0, 1]$ , indicating no rotation or rotation by angle of zero). An unbiased angular rate vector  $\omega_B$  which is in  $\mathbb{R}^3$  can be mapped into quaternion space ( $\mathbb{R}^4$ ) by augmenting zero as its real part, this quaternion is called a pure quaternion [15]. Since the real part of  $\omega_B$  in  $\mathbb{R}^4$  is zero, the quaternion multiplication can be written in matrix form as shown in Eq. (4).

$$\dot{q} = \frac{1}{2} \hat{q}_0 \otimes \omega_B = \frac{1}{2} \begin{bmatrix} 0 & \omega_{Bz} & -\omega_{By} & \omega_{Bx} \\ -\omega_{Bz} & 0 & \omega_{Bx} & \omega_{By} \\ \omega_{By} & -\omega_{Bx} & 0 & \omega_{Bz} \\ -\omega_{Bx} & -\omega_{By} & -\omega_{Bz} & 0 \end{bmatrix} \begin{bmatrix} \hat{q}_{0x} \\ \hat{q}_{0y} \\ \hat{q}_{0z} \\ \hat{q}_{0w} \end{bmatrix} \quad (4)$$

$$q_G = \exp((\Delta t)\dot{q} \otimes \hat{q}_0^*) \otimes \hat{q}_0 \quad (5)$$

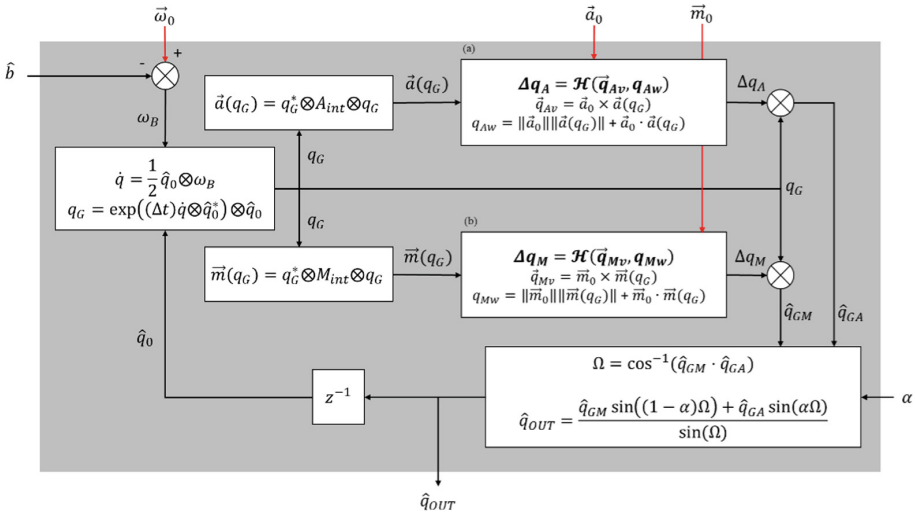
The quaternion representation of the orientation ( $q_G$ ) can be obtained by integrating the quaternion rate using Eq. (5), where  $\Delta t$  is the sampling period of the sensor's data. This quaternion ( $q_G$ ) can be used to rotate points or vectors in three-dimensional space. In other words, it can be used to describe the orientation of the sensor module or the object that the IMU is attached to. To justify the orientation estimation using this

method, we also proposed the method to compare this quaternion result with other referencing measurements from the accelerometer and magnetometer integrated in the same IMU.

### 2.2 Quaternion Correction

**Using the Gravity Vector.** Each of the three accelerometers in the IMU provides the value of acceleration applied to the sensor module in its sensing direction. When the sensor is static, the accelerometers are assumed to measure only the acceleration due to gravity, which is always directed towards the Earth’s center and has only a vertical component in the Earth frame. When the IMU module is in an oblique orientation (not at right angle or parallel with the horizon), the acceleration due to gravity is decomposed into each orthogonal axis in the sensor frame. These components can be represented as the inclination of the sensor module in the Earth frame. The triple-product quaternion operator shown in Eq. (6) is used to rotate the coordinate frame [15] with respect to vector  $\vec{v}$  into another coordinate frame represented by vector  $\vec{w}$  since  $q$  represents the orientation or the inclination of the sensor with respect to the Earth frame.

$$\vec{w} = q^* \otimes \vec{v} \otimes q \tag{6}$$



**Fig. 2.** Block diagram of the proposed drift correction algorithm using both (a) gravity vector and (b) Magnetic North vector correction

Accordingly, we can use the quaternion calculated from the integration of gyroscope reading from the first part ( $q_G$ ) to rotate the referencing frame of the gravity vector from the Earth frame into the sensor frame in order to obtain the gravity vector referencing in the sensor frame as shown in Eq. (7). This *calculated* gravity vector

$(\vec{a}(q_G))$  will further be used to compare with the *measured* gravity vector from the accelerometer readings.

$$\vec{a}(q_G) = q_G^* \otimes A_{int} \otimes q_G \quad (7)$$

If the estimated orientation of the IMU, represented by the quaternion  $q_G$  had no error,  $\vec{a}(q_G)$  from Eq. (7) would match the measured gravity vector from accelerometer readings. Otherwise, the error between these two vectors can be used to correct the orientation estimation by calculating the angular difference between these two vectors. The difference in angle of the calculated gravity vector ( $\vec{a}(q_G)$ ) and measured gravity vector from accelerometer readings ( $\vec{a}_0$ ) can be calculated and then represented as a rotation in quaternion form, as in Eq. (8). Then, the estimated quaternion using gravity vector correction ( $\hat{q}_{GA}$ ) is calculated by multiplying the quaternion from Eq. (5) with the difference in quaternion ( $\Delta q_A$ ) between the calculated gravity vector and the measured gravity vector from accelerometer readings, from Eq. (8). The estimated quaternion ( $\hat{q}_{GA}$ ) is described as in Eq. (11).

$$\Delta q_A = \mathcal{H}(\vec{q}_{Av}, q_{Aw}) \quad (8)$$

$$\vec{q}_{Av} = \vec{a}_0 \times \vec{a}(q_G) \quad (9)$$

$$q_{Aw} = \|\vec{a}_0\| \|\vec{a}(q_G)\| + \vec{a}_0 \cdot \vec{a}(q_G) \quad (10)$$

$$\hat{q}_{GA} = q_G \otimes \Delta q_A \quad (11)$$

This resulting quaternion  $\hat{q}_{GA}$ , which estimates the orientation of the sensor module by using gravity vector correction, will be used in the next step of the algorithm. Note that the quaternion result has to be normalized before proceeding with any calculations.

**Using the Magnetic North Vector.** Similar to quaternion correction using gravity vector, this part of the algorithm pursues a similar approach in using the other sensing source as the referencing vector. Unlike the gravity vector, the directions of the magnetic north vector at different locations on Earth are different. However, at the particular location, the magnetic north vector is always pointing towards the North magnetic pole. Therefore, each of the three magnetometers in the IMU should provide the value of magnetic field strength applied to the sensor module in its sensing direction. Whether the sensor is static or moving, the magnetometers are assumed to measure the direction of the magnetic North vector, which is always directed towards the North magnetic pole in the Earth's coordinate frame. When the IMU module is deviated from magnetic north vector direction (not parallel with the direction of the Earth magnetic field), the magnetic North vector is decomposed into each orthogonal axis in the sensor frame. These components can also be represented as the inclination of the sensor module in the Earth frame. Thus, we can use the quaternion calculated from the integration of gyroscope reading from the first part ( $q_G$ ) to rotate the referencing magnetic North vector in the Earth frame ( $M_{int}$ ) in order to obtain the magnetic North vector in the sensor frame as shown in Eq. (12). This *calculated* magnetic North vector

$(\vec{m}(q_G))$  will further be used to compare with the *measured* magnetic North vector from the magnetometer readings.

$$\vec{m}(q_G) = q_G^* \otimes M_{int} \otimes q_G \quad (12)$$

If the estimated orientation of the IMU, represented by the quaternion  $q_G$  had no error,  $\vec{m}(q_G)$  from Eq. (12) would match the measured magnetic North vector from magnetometer readings. Otherwise, the error between these two vectors can be used to correct the orientation estimation by calculating the angular difference between these two vectors. The difference in angle of the calculated magnetic North vector ( $\vec{m}(q_G)$ ) and measured magnetic North vector from magnetometer readings ( $\vec{m}_0$ ) can be calculated and then represented as a rotation in quaternion form, as in Eq. (13). Then, the estimated quaternion using gravity vector correction ( $\hat{q}_{GM}$ ) is calculated by multiplying the quaternion from Eq. (5) with the difference in quaternion ( $\Delta q_M$ ) between the calculated magnetic North vector and the measured magnetic North vector from magnetometer readings, from Eq. (13). The estimated quaternion ( $\hat{q}_{GM}$ ) is described as in Eq. (16).

$$\Delta q_M = \mathcal{H}(\vec{q}_{Mv}, q_{Mw}) \quad (13)$$

$$\vec{q}_{Mv} = \vec{m}_0 \times \vec{m}(q_G) \quad (14)$$

$$q_{Mw} = \|\vec{m}_0\| \|\vec{m}(q_G)\| + \vec{m}_0 \cdot \vec{m}(q_G) \quad (15)$$

$$\hat{q}_{GM} = q_G \otimes \Delta q_M \quad (16)$$

The resulting quaternion  $\hat{q}_{GM}$  estimates the orientation of the sensor module by using magnetic North vector correction and will proceed to the next step of the algorithm. Note that the quaternion result has to be normalized before proceeding with any calculations.

From Eq. (11),  $\hat{q}_{GA}$  is the product of quaternion multiplication between  $q_G$  and  $\Delta q_A$  which represents the combination of both rotations. Since  $\Delta q_A$  is the rotation that represents angular difference in quaternion form between the calculated gravity vector and the measured gravity vector from accelerometer readings, the quaternion product of  $q_G \otimes \Delta q_A$  then represents the estimated orientation that is fully corrected based on the measurement of accelerometer. Similarly,  $\hat{q}_{GM}$  also represents the estimated orientation that is fully corrected based on the measurement of magnetometer.

### 2.3 Quaternion Interpolation

With the combination of bias offset estimation and quaternion correction using gravity vector, the algorithm can be used to correct the drift in rotation and improve the orientation estimates [16]. However, when a rapid movement of the hand occurs, the accelerometer measures not only just acceleration due to gravity but also the acceleration due to linear motion. Thus, the accelerometer readings can no longer be used as a reliable source of reference. In contrast, the magnetometer readings are not affected by the linear acceleration due to rapid movement of the hand (However, the magnetic field

perceived by the sensor may not be uniform around the user, due to the presence of ferromagnetic objects nearby). The periods of time when the hand is more likely to have rapid translation and linear acceleration inclusion are the moments when the orientation estimates should rely more on the orientation estimates using magnetometer ( $\hat{q}_{GM}$ ) than the orientation estimates using accelerometer ( $\hat{q}_{GA}$ ). An appropriate method to partially or selectively combine two different rotations by using a parametric weight is the method used in computer graphic animation called Spherical Linear Interpolation (SLERP) or great arc interpolation [17].

$$SLERP(q_0, q_1, h) = \frac{q_0 \sin((1-h)\Omega) + q_1 \sin(h\Omega)}{\sin(\Omega)} \quad (17)$$

$$\cos(\Omega) = q_0 \cdot q_1 \quad (18)$$

Spherical Linear Interpolation of two quaternions as shown in Eq. (17) is the method in animation to calculate the quaternion or the rotation representation in the intermediate stages between the starting ( $q_0$ ) and the ending ( $q_1$ ) of rotation sequence. The control parameter ( $h$ ) that interpolates the rotation ranges from 0 to 1, where the control parameter that tends towards zero interpolates the rotation close to the starting rotation, and the control parameter that tends towards one interpolates the rotation close to the ending rotation. The control parameter divides the interpolated rotation linearly. Evidently, the Spherical Linear Interpolation preserves the normalization of the quaternion and all the interpolated rotations lie on the same great arc.

In Eq. (19), the Spherical Linear Interpolation method was adapted to determine the final estimated orientation ( $\hat{q}_{OUT}$ ) between the orientation estimate using magnetometer ( $\hat{q}_{GM}$ ) and the orientation estimate using accelerometer ( $\hat{q}_{GA}$ ). The angle of the interpolated arc subtended by  $\hat{q}_{GM}$  and  $\hat{q}_{GA}$  is denoted by  $\Omega$ , where  $\cos(\Omega)$  equals to the 4-dimensional inner product of  $\hat{q}_{GM}$  and  $\hat{q}_{GA}$ . The control parameter of quaternion interpolation ( $\alpha$ ) is the parameter that indicates the stillness of the sensor module. When  $\alpha$  equals to 1, it indicates a static period of the sensor module. Thus, the final estimated quaternion orientation is solely estimated using accelerometer data ( $\hat{q}_{GA}$ ) since the accelerometer is assumed to measure only the acceleration due to gravity when the sensor is static. When there is a movement, the accelerometer measurement includes acceleration due to linear motion, resulting in the drop of stillness. Consequently, the control parameter of quaternion interpolation ( $\alpha$ ) is close to zero. In this case, the final estimated quaternion is the interpolated orientation estimate that tends towards the orientation estimates using magnetometer ( $\hat{q}_{GM}$ ).

$$\hat{q}_{OUT} = \frac{\hat{q}_{GM} \sin((1-\alpha)\Omega) + \hat{q}_{GA} \sin(\alpha\Omega)}{\sin(\Omega)} \quad (19)$$

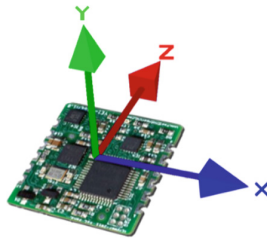
$$\cos(\Omega) = \hat{q}_{GM} \cdot \hat{q}_{GA} \quad (20)$$

This algorithm to estimate the orientation in quaternion form is also visually described in the block diagram shown in Fig. 2. The final orientation estimates ( $\hat{q}_{OUT}$ )

is also fed back for the calculation of the quaternion estimates in the initial stage ( $q_G$ ) for the next iteration.

### 3 Implementation

In this work, we used two 3-Space Sensor modules, from Yost Labs. This module is an ultra-miniature, high-precision, high-reliability, low-cost Inertial Measurement Unit (IMU). The module has the dimensions of 23 mm  $\times$  23 mm  $\times$  2.2 mm (0.9  $\times$  0.9  $\times$  0.086 in.) and consists of tri-axial gyroscope, accelerometer, and magnetometer. The coordinate axes of the sensor's frame are superimposed on the image of the sensor's module shown in Fig. 3.



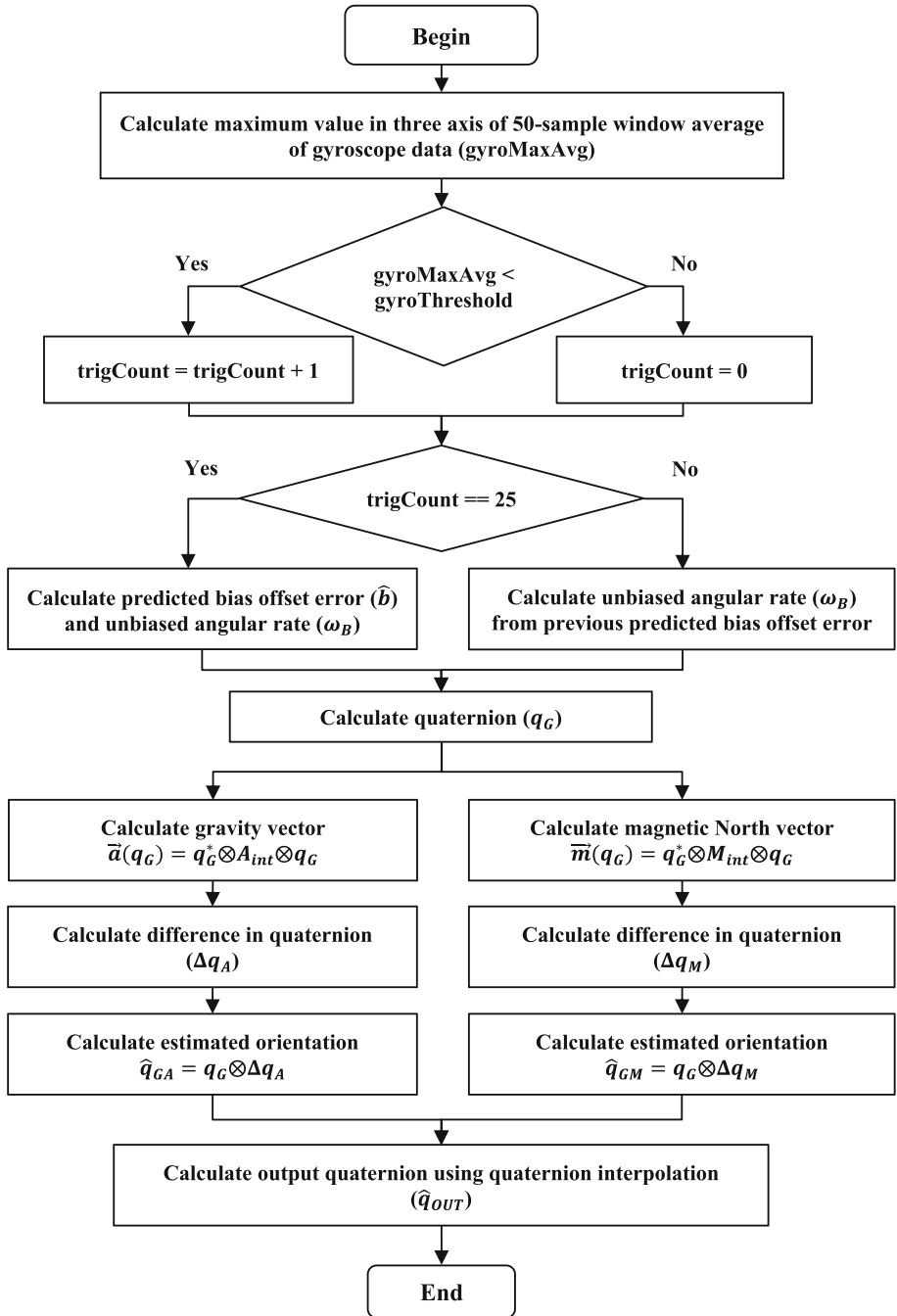
**Fig. 3.** Coordinate axes superimposed on the image of YEI 3-Space Embedded IMU module

These Yost Labs 3-Space Sensor modules were attached on the back of the hand and on the tip of the index finger as shown in Fig. 4. The gyroscope, accelerometer and magnetometer data were recorded and stored in a text file while a sequence of hand motions were being performed. The orientation correction algorithm was applied to the recorded data within MATLAB. In order to be able to effectively apply this orientation correction algorithm to the hand motion tracking system, the algorithm are supposed to work in real-time. Therefore, during the implementation of the algorithm in MATLAB, the programming code was aware of only one sample of data at a time in order to imitate real-time performance.



**Fig. 4.** The glove with IMUs attached on the back of the hand and on the tip of index finger





**Fig. 5.** Flowchart showing the implementation of orientation correction algorithm for one iteration using both gravity and magnetic North vector corrections.

The bias offset error must be estimated when the IMUs detect periods of stillness. A flowchart showing the implementation of the orientation correction algorithm is displayed in Fig. 5. It shows that the algorithm begins with the calculation of the maximum value in a 50-sample window average of gyroscope data (from the 3 axes) called *gyroMaxAvg*.

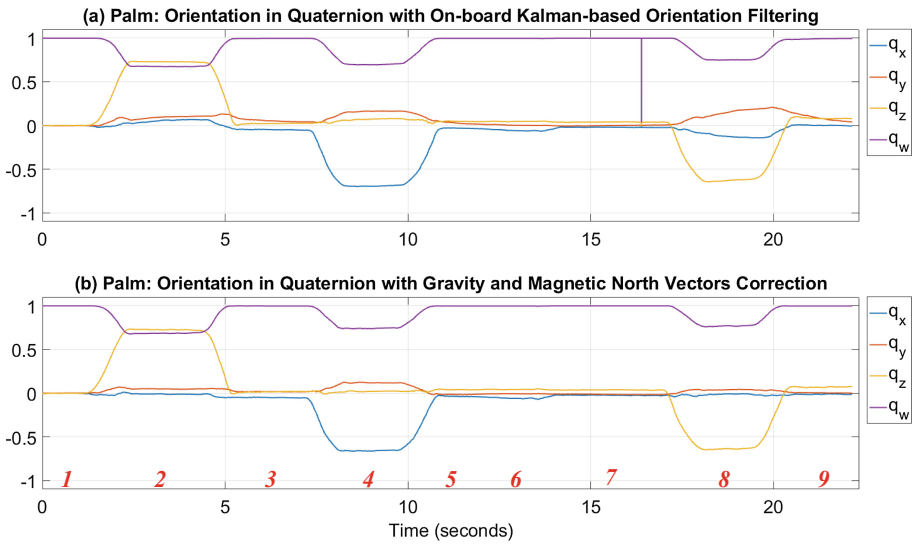
If this value is less than a pre-defined threshold value (*gyroThreshold*), that IMU is considered to be in the static period (this condition has to be true for 25 consecutive samples). Then, the new predicted bias offset error will be calculated. Otherwise, the previous predicted bias offset error will be used to calculate the unbiased angular rate instead. The quaternion  $q_G$  is calculated by integrating the unbiased angular rate. The quaternion correction using gravity-vector and magnetic North vector are implemented simultaneously. The output quaternion ( $\hat{q}_{OUT}$ ) is calculated by using quaternion interpolation between estimated quaternion correction using gravity vector ( $\hat{q}_{GA}$ ) and estimated quaternion correction using magnetic North vector ( $\hat{q}_{GM}$ ). In our experiments, the resulting orientations were exported as a text file and visualized by the animation of a 3D hand model in Unity. The motion of the hand model with corrected orientation was validated and compared with snapshots of the hand motion sequence.

## 4 Results and Discussion

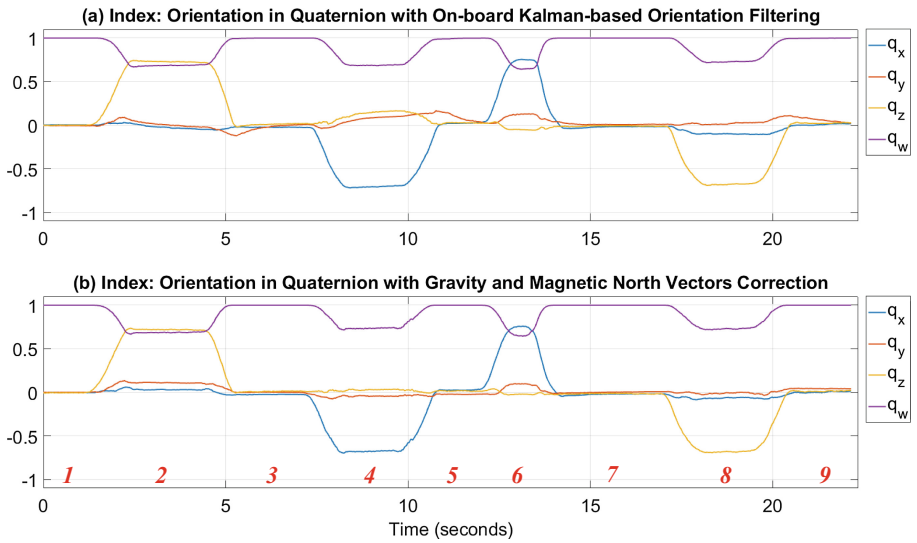
Figures 6(a) and 7(a) show the orientation in quaternion for IMUs attached on the back of the hand and the index finger. They were recorded directly from the sensor modules for a duration of 22.348 s. These recordings are the quaternion output from the Kalman-based orientation filtering inside the Yost Lab's 3-space sensor module.

The resulting estimated orientation after gravity-vector and magnetic North vector correction for IMUs attached on the back of the hand and the index finger are shown in Figs. 6(b) and 7(b), respectively. Comparing the quaternion results from the orientation correction algorithm using gravity and magnetic North vector and the quaternion recording from the on-board Kalman-based filtering in Fig. 6, it was found that the result from the algorithm using gravity and magnetic North vector correction can reduce the amount of drift that distinctly occurred in Kalman-based filtering estimation during the 2<sup>nd</sup> to 5<sup>th</sup> seconds and 17<sup>th</sup> to 21<sup>st</sup> seconds. Similarly in Fig. 7, the quaternion result of the IMU attached on the index finger from the orientation correction algorithm using gravity and magnetic North vector and the quaternion recording from the on-board Kalman-based filtering are compared. This figure shows that the result from the algorithm using gravity and magnetic North vector correction can also reduce the amount of drift in the quaternion result using Kalman-based filtering method that occurred during the 7<sup>th</sup> to 11<sup>th</sup> seconds, 12<sup>th</sup> to 14<sup>th</sup> seconds and 17<sup>th</sup> to 21<sup>st</sup> seconds of the recording.

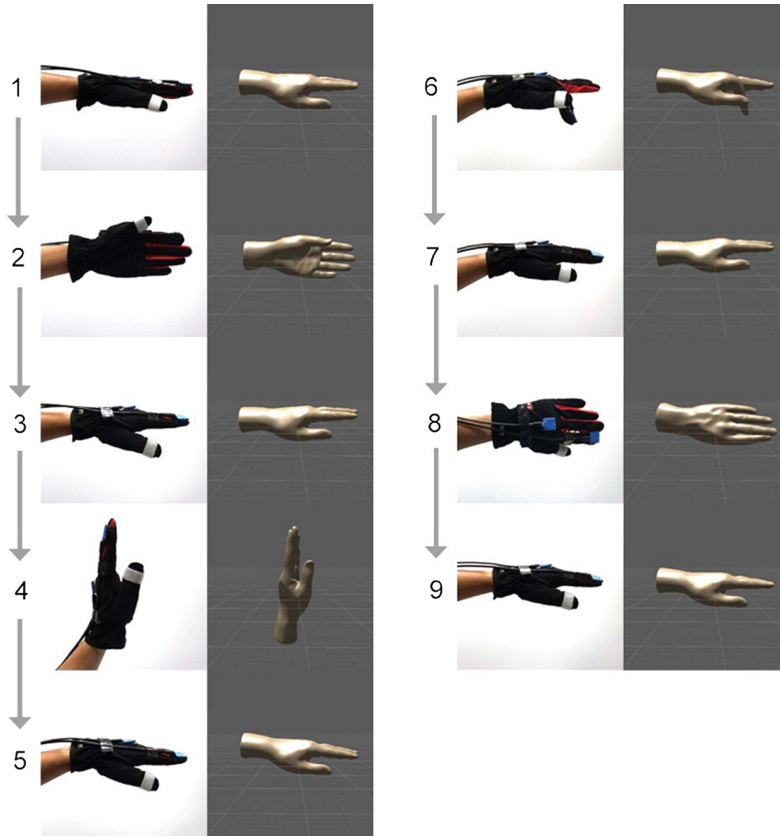
To verify these results, the quaternion outputs obtained from the gravity and magnetic North correction algorithm were exported as text files and were applied to the rotation of the 3D hand model in Unity. Figure 8 shows the comparison between a sequence of actual photographs of the hand orientation and 3D visualizations of estimated hand orientation after gravity-vector and magnetic North vector correction. Comparing the results to the actual hand orientation, leads us to confirm that the



**Fig. 6.** The plots of estimated quaternion results (a) with On-board Kalman-based Orientation Filtering and (b) with gravity and magnetic North vectors correction for IMUs attached on the back of the hand



**Fig. 7.** The plots of estimated quaternion results (a) with On-board Kalman-based Orientation Filtering and (b) with gravity and magnetic North vectors correction for IMUs attached on the index finger



**Fig. 8.** Comparison between a sequence of actual hand orientations and 3D visualizations of estimated hand orientation after gravity-vector and magnetic North vector correction.

estimated quaternion after gravity-vector and magnetic North vector correction provides an acceptable result of orientation tracking using the IMUs. The 9-stage sequences of 3D hand orientation in Fig. 8 are the visualization of the quaternion results in Figs. 6(b) and 7(b). The numbers (1–9) to the left of the stages in Fig. 8 correspond with the (1–9) intervals labelled at the bottom of Figs. 6(b) and 7(b).

## 5 Conclusion

We believe that our approach to correct the drift in the gyroscope measurements and compensate the orientation estimation using the gravity vector and magnetic North vector will be useful in orientation tracking of human hand motion. Since the precise data of the hand or the fingers orientation can be obtained, this can lead to the improvement of 3D user interfaces to become more realistic. Thus, it could also enhance the experiences of the natural interaction of humans with a 3D virtual environment.

**Acknowledgment.** This research was supported by National Sciences Foundation grants HRD-0833093 and CNS-1532061 and the FIU Graduate School Dissertation Year Fellowship awarded to Mr. Nonnarit O-larnnithipong.

## References

1. McCall, R., et al.: Measuring Presence in Virtual Environments. ACM (2004)
2. Slater, M., Usoh, M., Steed, A.: Depth of presence in virtual environments. *Presence Teleoperators Virtual Environ.* **3**(2), 130–144 (1994)
3. Slater, M., Wilbur, S.: A framework for immersive virtual environments (FIVE). *Presence Teleoperators Virtual Environ.* **6**(6), 603 (1997)
4. Heeter, C.: Being there: the subjective experience of presence. *Presence Teleoperators Virtual Environ.* **1**(2), 262–271 (1992). <https://doi.org/10.1162/pres.1992.1.2.262>
5. Sukkarieh, S., Nebot, E.M.: A high integrity IMU/GPS navigation loop for autonomous land vehicle applications. *IEEE Trans. Robot. Autom.* **15**(3), 572 (1999)
6. Borenstein, J., Everett, H.R., Feng, L., et al.: *Mobile Robot Positioning Sensors and Techniques*. Naval Command Control and Ocean Surveillance Center RDT and E Div, San Diego (1997)
7. Marins, J.L., Yun, X., Bachmann, E.R., et al.: An extended Kalman filter for quaternion-based orientation estimation using MARG sensors. In: 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems, vol. 4, pp. 2003–2011. IEEE (2001)
8. Yun, X., Bachmann, E.R.: Design, implementation, and experimental results of a quaternion-based kalman filter for human body motion tracking. *IEEE Trans. Robot.* **22**(6), 1216–1227 (2006). <https://doi.org/10.1109/tro.2006.886270>
9. Yun, X., Lizzarraga, M., Bachmann, E.R., et al.: An improved quaternion-based Kalman filter for real-time tracking of rigid body orientation. In: 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems, vol. 2, pp. 1074–1079. IEEE (2003)
10. Madgwick, S.O., Harrison, A.J., Vaidyanathan, R.: Estimation of IMU and MARG orientation using a gradient descent algorithm. In: 2011 IEEE International Conference on IEEE Rehabilitation Robotics (ICORR), pp. 1–7 (2011)
11. Madgwick, S.: An efficient orientation filter for inertial and inertial/magnetic sensor arrays. Report x-io and University of Bristol, UK, 25 (2010)
12. Bachmann, E.R., Duman, I., Usta, U.Y., et al.: Orientation tracking for humans and robots using inertial sensors. In: 1999 IEEE International Symposium on IEEE Computational Intelligence in Robotics and Automation, CIRA 1999, Proceedings, pp. 187–194 (1999)
13. Kong, X.: INS algorithm using quaternion model for low cost IMU. *Robot. Auton. Syst.* **46**(4), 221–246 (2004)
14. Montgomery, D.C., Corporation, E., Peck, E.A., et al.: *Introduction to Linear Regression Analysis*, 5th edn. Wiley, Hoboken (2012)
15. Kuipers, J.B.: *Quaternions and Rotation Sequences: A Primer with Applications to Orbits, Aerospace, and Virtual Reality*. Princeton University Press, Princeton (1999)
16. O-larnnithipong, N., Barreto, A.: Gyroscope drift correction algorithm for inertial measurement unit used in hand motion tracking. In: 2016, IEEE SENSORS, pp. 1–3 (2016)
17. Dam, E.B., Koch, M., Lillholm, M.: *Quaternions, Interpolation and Animation*. Datalogisk Institut, Kbenhavns Universitet (1998)