

The 3D Interaction Tool: A Pointing Device for Virtual Reality Applications

Lukas Gemar

March 26, 2016

Abstract

The 3D Interaction Tool is a hand-held device that enables a user to interact with virtual reality applications. Since both the position and orientation of the tool are tracked, user input with the tool accomplishes tasks such as selecting, translating, and rotating three-dimensional objects, or free-hand drawing in space. The tools position is tracked by a software application that filters observations from a camera attached to the virtual reality display, and the orientation is found by means of observations from an inertial measurement unit attached to the tool itself. Simple computer-aided design and drawing applications demonstrate the capabilities of the 3D Interaction Tool.

I. Nomenclature

Position vectors definitions in the inertial coordinate frame:

\vec{p}_s \equiv position of the spherical tracking target on tool rear

\vec{p}_r \equiv position of the tool rear

\vec{p}_f \equiv position of the tool front

\vec{p}_c \equiv position of the tracking camera

\vec{p}_d \equiv position of the virtual reality display

Rotated position vectors:

\vec{p}'_s \equiv position of the spherical tracking target on tool rear, camera coordinates

\vec{p}'_r \equiv position of the tool rear in display coordinates

\vec{p}'_f \equiv position of the tool front in display coordinates

Attitude matrices:

\mathbf{A}_c \equiv attitude of the tracking camera

\mathbf{A}_d \equiv attitude of the virtual reality display

\mathbf{A}_t \equiv attitude of the interaction tool

\mathbf{A}'_t \equiv attitude of the interaction tool in display coordinates

Body-referenced quaternion representations of attitude:

$\bar{\mathbf{q}}_c$ \equiv attitude of the tracking camera

$\bar{\mathbf{q}}_d$ \equiv attitude of the virtual reality display

$\bar{\mathbf{q}}_t$ \equiv attitude of the interaction tool

Position error covariances in inertial coordinate frame:

\mathbf{P}_{ps} \equiv covariance of spherical tracking target position

\mathbf{P}_{pr} \equiv covariance of the position of the rear of the interaction tool

\mathbf{P}_{pf} \equiv covariance of the position of the front of the interaction tool

\mathbf{P}_{pc} \equiv covariance of the position of the tracking camera

\mathbf{P}_{pd} \equiv covariance of the position of the virtual reality display

Attitude error covariances in the inertial coordinate frame:

\mathbf{P}_{ac} \equiv covariance of the tracking camera attitude quaternion error

\mathbf{P}_{ad} \equiv covariance of the display attitude quaternion error

\mathbf{P}_{at} \equiv covariance of the interaction tool attitude quaternion error

Position error covariances in rotated body coordinates:

\mathbf{P}'_{ps} \equiv covariance of the spherical tracking target position

\mathbf{P}'_{pr} \equiv covariance of the position of the rear of the interaction tool

\mathbf{P}'_{pf} \equiv covariance of the position of the front of the interaction tool

Attitude error covariances in the rotated body coordinates:

\mathbf{P}'_{ac} \equiv covariance of the tracking camera attitude angular error

\mathbf{P}'_{ad} \equiv covariance of the display attitude angular error

\mathbf{P}'_{at} \equiv covariance of the interaction tool attitude angular error

II. Introduction

A. Motivation

As much as \$25 billion could be spent on virtual reality (VR) and augmented reality (AR) software applications by 2025 [1]. There are two primary modes of user-input that are possible for these

applications: indirect and direct. Indirect input devices translate the user's physical movements into the motion of a virtual cursor by means of a fixed transformation. Common examples of indirect input devices are a mouse or joystick. Direct input devices also translate physical to virtual movements, but this transformation is one-to-one. For example, a touch screen is a direct input device because the physical location of the user's press is colocated with the virtual click.

With the advent of virtual reality applications the following question must be answered: which of these input paradigms – indirect or direct – will prove most useful for interacting with virtual reality applications? While indirect input devices have achieved widespread popularity for two-dimensional displays, the direct input paradigm for VR applications embraces the immersiveness of VR experiences by enabling intuitive interactions for users. Direct input devices allow users to reach out and interact with the objects of their virtual space, just as they would with objects in their physical space.

B. Background

C. Usability Objectives

The usability requirements presented in Figure 1 must be met for the tool to succeed as an *input* device for VR software applications.

Usability		
Objective	Justification	Target
Good portability	Target scenarios are seated applications for home and workplace.	The interaction tool setup should be easily translatable between uses.
Low configuration overhead	---	Limited setup time.
Natural form factor	The interaction tool should not require the user to learn a new interaction paradigm.	Form factor similar to a pen or stylus.
Large tracking volume	The user should be able to use the interaction tool over their full range of movement.	Tracking volume of about 1m^3 .
Low cost	---	Less than \$500.

Figure 1: Usability design objectives

D. Performance Objectives

The performance requirements presented in Figure 2 must be met for the tool to succeed as a *direct* input device for VR software applications.

Performance		
Objective	Justification	Target
High Accuracy	Tracking apparatus must translate user movements into direct input signals.	10cm or less.
Very High Precision	High precision ensures that cursor movements are repeatable.	1-2cm.
Low Latency	Too much discrepancy between physical and virtual motion of the tool induces simulator sickness.	Less than 60ms.
High Robustness	The system should be usable in non-ideal environments and use-cases.	Robust to tool velocity.

Figure 2: Performance design objectives

III. Tracking System Design

A. Requirements

In order to enable direct input with the interaction tool, the tool's tracking system must estimate both the position and orientation of the interaction tool. Additionally, rendering a virtual representation of the tool requires measuring its position and orientation relative to the virtual reality display orientation, given by the attitude matrix \mathbf{A}_d . To track the tool's absolute trajectory, its position is determined relative to the world – that is, relative to a non-accelerating, inertial coordinate system in the user's physical environment. The position and orientation of the tool in the world coordinates of the inertial frame are \vec{p}_f and \mathbf{A}_t , respectively. The position of the tool in the coordinate frame of the virtual reality display is denoted by a 'prime' sign: the tip or front of the tool in display coordinates is \vec{p}'_f . Thus, there are four tracking targets: \vec{p}_f , \mathbf{A}_t , \vec{p}'_f , and \mathbf{A}_d ; these are shown in Figure 3.

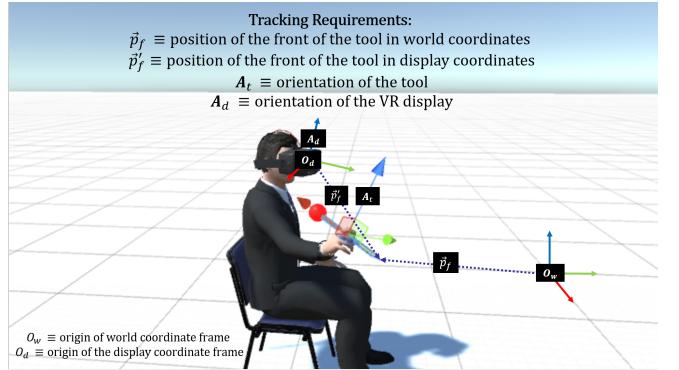


Figure 3: Position and orientation tracking requirements

B. Geometry

The geometry of the tool tracking system affects both the usability and the performance of the interaction tool. The display-referenced geometry is chosen because it eliminates the burden on the user to pre-calibrate their environment before providing direct input to VR applications. At the same time, this geometry is compatible with VR applications with which a user interacts while seated.

B.1 Option 1: World-referenced geometry

The world-referenced tracking geometry is shown in Figure 4.

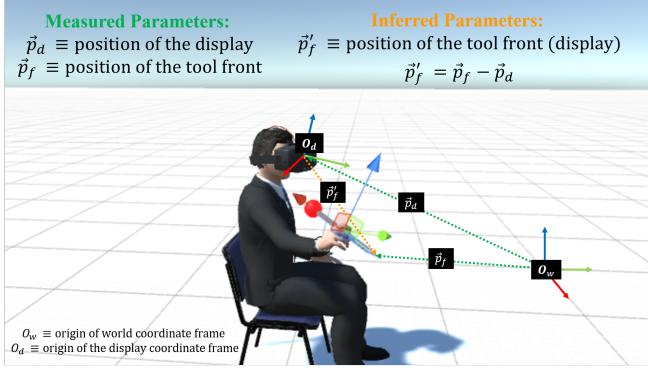


Figure 4: Position tracking with a world-referenced geometry

B.2 Option 2: Display-referenced geometry

The display-referenced tracking geometry is shown in Figure 5.

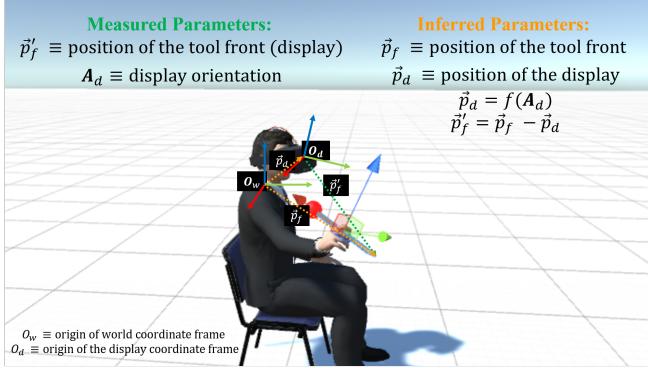


Figure 5: Position tracking with a display-referenced geometry

B.3 Geometry Selection: Comparing portability and tracking volume

Tracking geometries 1 and 2 have advantages and disadvantages when evaluated using the usability design objectives. Relative to the usability criteria, the two geometries differ greatly in their portability and tracking volume.

The world-referenced tracking geometry requires a fixed coordinate system in the user's physical environment, decreasing portability by increasing the configuration overhead for using the direct input device. However, once this coordinate system is established, the user can provide direct input to a VR application over a very large tracking volume, up to the size of the room in which the tracking system is located. The display-referenced geometry, on the other hand, has the advantage of enabling a user to

provide direct input to a VR application without pre-calibrating the physical environment in which they are using the VR display. The tradeoff is that the tracking volume is small. Since the origin of the world coordinate system in the display-referenced geometry is located beneath the user's head, the user must remain seated while using a direct input device with a display-referenced tracking geometry.

C. Technology

A combination between optical and inertial sensors was chosen to track the 3D Interaction Tool's position and orientation. The chart in Figure 6 summarizes the comparison between various alternatives for the choice of the tracking technology.

Tracking Technology	Advantages	Disadvantages
Mechanical	Very high accuracy and precision . Good portability .	Small tracking volume .
Magnetic	Very high accuracy and precision . Low latency . Good robustness .	Very high cost .
Ultrasound	High accuracy and precision .	Poor portability . Incompatible with slender form factor. Poor robustness to acoustic noise.
Optical	High accuracy and precision . Low configuration overhead . Low cost .	Poor robustness to occlusions and tool velocity. High latency . Narrow field of view .
Inertial	High accuracy and precision of orientation. Very low latency . Low cost .	Low accuracy and precision of position over large time scales.

Figure 6: Comparison of five tracking technologies

D. Final Design

D.1 Tracking measurements

Not all of these three tracking targets are measured directly. The tracking targets are inferred from the measurements of the position of the rear of the interaction tool relative to the camera, the attitude of the camera, and the attitude of the tool. The position of the tool relative to the camera is denoted by \vec{p}'_s , the attitude of the camera relative to the inertial coordinate system by \mathbf{A}_c and the attitude of the tool relative to the inertial coordinate system by \mathbf{A}_t . The vector \vec{p}'_s is measured by finding the .

D.2 Inferring the tool position

D.3 Inferring the tool attitude

$$\vec{p}_r = \mathbf{A}_c^{-1} \vec{p}'_s - \vec{p}_c$$

$$\vec{p}_f = \vec{p}_r + \mathbf{A}_t^{-1} \vec{l}$$

$$\vec{p}'_f = \mathbf{A}_d(\vec{p}_f + \vec{p}_d)$$

$$\mathbf{A}'_t = \mathbf{A}_d \mathbf{A}_t^{-1} \vec{l}$$

E. Error Analysis

E.1 Position estimation error

E.2 Orientation estimation error

E.3 Model error

F. Experimental Setup

IV. Position Tracking

A. Measurement Model

The following is a model of a single frame from the camera. In this frame, shown in image Figure 7 the red circular object represents the back of the interaction tool. The back of the interaction tool is called the target. The width of the target along the x dimension of the image is d_u .

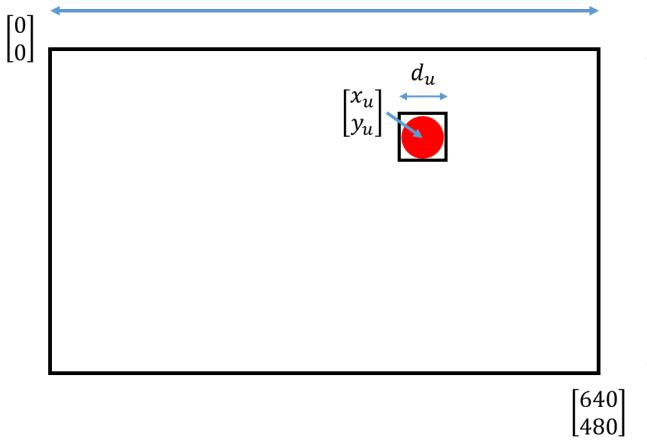


Figure 7: Computer vision tracking model

The midpoint of the target in the ideal undistorted image is $\mathbf{x}_E^u = \begin{bmatrix} x_u & y_u \end{bmatrix}^T$. The center of the undistorted image is c_x in the x direction and c_y in the y direction. The focal lengths in the x and y directions are f_x and f_y , respectively. These parameters are found through the camera calibration routine. The measurements of the center of the target and its width are related to the hidden state $\mathbf{x}_C = \begin{bmatrix} x_C & y_C & z_C \end{bmatrix}^T$ in the following way:

$$\mathbf{y} = \mathbf{h}(\mathbf{x}) = \begin{bmatrix} x_u \\ y_u \\ d_u \end{bmatrix} = \begin{bmatrix} \frac{f_x x_c}{z_c} + c_x \\ \frac{f_y y_c}{z_c} + c_y \\ \frac{d_c}{z_c \sqrt{\frac{1}{f_x^2} + \frac{1}{f_y^2}}} \end{bmatrix}$$

These relationships are found through using a simple pin-hole camera model [2].

B. Camera Calibration

The measurement model requires knowledge of the intrinsic camera parameters, f_x, f_y, c_x, c_y , and the radial and tangential distortion factors. There are two steps to translating a point from camera coordinates to image coordinates [2]. First, the pin-hole camera model shows how to find the ideal image coordinates of an object, $(\mathbf{x}_u, \mathbf{y}_u)$ based on the position of the object in camera coordinates, (x, y, z) :

$$\mathbf{x}_u = f_x \frac{\mathbf{x}}{z}$$

$$\mathbf{y}_u = f_y \frac{\mathbf{y}}{z}$$

Second, the radial and tangential distortion relations show how the actual image coordinates, $(\mathbf{x}_d, \mathbf{y}_d)$, relate to the image coordinates on the actual or distorted image plane, $(\mathbf{x}_d, \mathbf{y}_d)$. The radial distortion correction equation is as follows:

$$\mathbf{x}_u = \mathbf{x}_d (1 + k_1 r^2 + k_2 r^4 + k_3 r^6)$$

$$\mathbf{y}_u = \mathbf{y}_d (1 + k_1 r^2 + k_2 r^4 + k_3 r^6)$$

and the tangential distortion equation is given by,

$$\mathbf{x}_u = \mathbf{x}_d + (2p_1 xy + p_2(r^2 + 2x^2))$$

$$\mathbf{y}_u = \mathbf{y}_d + (p_1(r^2 + 2y^2) + 2p_2xy)$$

Thus, the five distortion parameters are summarized in a vector D :

$$D = \begin{bmatrix} k_1 & k_2 & p_1 p_2 & k_3 \end{bmatrix}$$

The calibration parameters are discovered by using an open source implementation provided by OpenCV. The camera collects a series of images that contain a checkerboard pattern. The calibration computes the transformation from the image coordinates of the checkerboard to its real world coordinates. The calibration images look like the ones in figure 1 below.

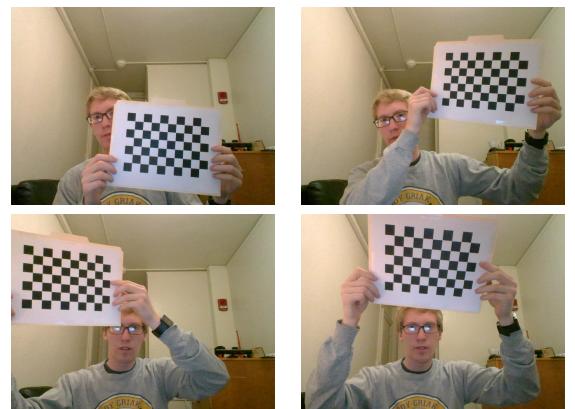


Figure 8: Camera calibration routine

V. Orientation Tracking

The orientation of the interaction tool relative to the world coordinate frame can be represented in a number of ways. The two representations that will be important here are the attitude matrix and the attitude quaternion. First, the attitude of the orientation tool can be described by the attitude matrix A . The attitude matrix typically maps a vector in the reference frame to a vector in the body frame [3]. This mapping is given by the relation,

$$\mathbf{b} = \mathbf{Ar}$$

Second, the attitude of the interaction tool can be represented as a quaternion. A rotation in three-dimensions is equivalent to an axis of rotation and angle of rotation. If \hat{n} is the axis of a rotation and θ is the angle of the rotation, then the quaternion \vec{q} represents the attitude:

$$\vec{q} = \begin{bmatrix} \vec{q} \\ q_4 \end{bmatrix}$$

with $\vec{q} = \hat{n} \sin \frac{\theta}{2}$ and $q_4 = \cos \frac{\theta}{2}$. The quaternion must have length 1: $\vec{q}^T \vec{q} = 1$.

The attitude matrix and the quaternion representing attitude are related by the following equation, where A is a function of \vec{q} [4]:

$$A(\vec{q}) = (q_4^2 - \|\vec{q}\|^2)I_{3 \times 3} + 2\vec{q}\vec{q}^T - 2q_4[\vec{q} \times]$$

where $[\vec{q} \times]$ is the skew-symmetric cross product matrix,

$$[\vec{q} \times] = \begin{bmatrix} 0 & -q_3 & q_2 \\ q_3 & 0 & -q_1 \\ -q_2 & q_1 & 0 \end{bmatrix}$$

The relationship between A and \vec{q} can also be expressed as the product of two matrices [3]:

$$A(\vec{q}) = \Xi(\vec{q})^T \Psi(\vec{q})$$

with

$$\Xi(\vec{q}) = \begin{bmatrix} q_4 \mathbf{I}_{3 \times 3} + [\vec{q} \times] \\ -\vec{q}^T \end{bmatrix}$$

$$\Psi(\vec{q}) = \begin{bmatrix} q_4 \mathbf{I}_{3 \times 3} - [\vec{q} \times] \\ -\vec{q}^T \end{bmatrix}$$

A. TRIAD Attitude Estimation

The TRIAD estimation method finds the attitude matrix A that maps three reference frame vectors \vec{r}_k to their corresponding body frame vectors \vec{b}_k for $k = 1, 2, 3$ [5]. Two observations of reference frame vectors provide sufficient information to compute a right-handed orthonormal triad of vectors in the reference frame. If \vec{v}_1 and \vec{v}_2 are reference frame vectors for the accelerometer and the magnetometer, respectively, then

$$\vec{v}_1 = \begin{bmatrix} 0 \\ 0 \\ 9.81 \end{bmatrix} \quad \text{and} \quad \vec{v}_2 = \begin{bmatrix} 0.3 \\ -1 \\ 0 \end{bmatrix}$$

so that the gravity vector points down and the magnetometer vector points in the direction of magnetic North in Boston, relative to the world reference frame. The orthonormal vector triad representing the reference frame is given by,

$$\vec{r}_1 = \frac{\vec{v}_1}{\|\vec{v}_1\|}, \quad \vec{r}_2 = \frac{\vec{v}_1 \times \vec{v}_2}{\|\vec{v}_1 \times \vec{v}_2\|}, \quad \vec{r}_3 = \vec{r}_1 \times \vec{r}_2$$

Let \vec{w}_1 be the measurement of the normalized unit vector from the accelerometer and \vec{w}_2 be the measurement of the normalized unit vector from the magnetometer, then the orthonormal vector triads for the reference vectors are given by,

$$\vec{b}_1 = \frac{\vec{w}_1}{\|\vec{w}_1\|}, \quad \vec{b}_2 = \frac{\vec{w}_1 \times \vec{w}_2}{\|\vec{w}_1 \times \vec{w}_2\|}, \quad \vec{b}_3 = \vec{w}_1 \times \vec{w}_2$$

If there was no noise in the system, then it would be the case that $\vec{b}_k = A\vec{r}_k$ for $k = 1, 2, 3$. An equivalent way of writing this in matrix form is the following:

$$M_b = AM_r$$

where

$$M_b = \begin{bmatrix} \vec{b}_1 & \vec{b}_2 & \vec{b}_3 \end{bmatrix}$$

$$M_r = \begin{bmatrix} \vec{r}_1 & \vec{r}_2 & \vec{r}_3 \end{bmatrix}$$

Since M_r is an orthogonal matrix, its inverse is equal to its transpose: $M_r^{-1} = M_r^T$. Thus, the solution for the attitude matrix is,

$$A = M_b M_r^T$$

B. Dynamic Model for Rotations

The rate of change of the attitude can be written in the form of a general state space model:

$$\dot{\vec{x}} = A(\vec{x}, t)\vec{x} + B(\vec{x}, t)\vec{u}$$

The kinematic equation for the change in attitude as a function of the angular rate can be written as a rate of change of the attitude matrix or a rate of change in the quaternion [4]. The angular velocity vector is given by $\vec{\omega}$. The rate of change of the attitude matrix is,

$$\dot{A}(t) = [\vec{\omega} \times] A(t)$$

In this equation, $[\vec{\omega} \times]$ is an instance of the state transition matrix $A(\vec{x}, t)$ from the general schema for a state space model. The kinematic relation for the rate of change of the quaternion can be written in the following two ways:

$$\dot{\bar{q}} = \frac{1}{2}\Omega(\vec{\omega})\bar{q}$$

$$\dot{\bar{q}} = \frac{1}{2}\Xi(\bar{q})\vec{\omega}$$

where the first relation is a state space model with state transition matrix $A(\vec{\omega}) = \Omega(\vec{\omega})$, where the state transition matrix is a function of the angular rate vector $\vec{\omega}$:

$$\Omega(\vec{\omega}) = \begin{bmatrix} -[\vec{\omega} \times] & \vec{\omega} \\ -\vec{\omega}^T & 0 \end{bmatrix}$$

The second kinematic equation for the rate of change of the quaternion is an instance of the schema for the state space model where the angular rate $\vec{\omega}$ is treated as the input \vec{u} to the system and the input matrix $B(\bar{q}) = \Xi(\bar{q})$ is a function of the current attitude \bar{q} .

There are two good reasons to use the quaternion representation of attitude: first, the rate of change of the quaternion is linearly related to the state of the quaternion; and second, successive rotations are computed by taking the product of quaternions [3]. The following identity holds of the composition of quaternions [4]:

$$A(\bar{q}_1)A(\bar{q}_2) = A(\bar{q}_1 \otimes \bar{q}_2)$$

with the operation $A(\bar{q})$ finding the attitude matrix A that corresponds with the quaternion \bar{q} . Following the notation in [3], the product of the quaternion is linear in each of the quaternions \bar{q}_1 and \bar{q}_2 :

$$\bar{q}_1 \otimes \bar{q}_2 = \begin{bmatrix} \Psi(\bar{q}_1) & \bar{q}_1 \end{bmatrix} \bar{q}_2 = \begin{bmatrix} \Xi(\bar{q}_1) & \bar{q}_1 \end{bmatrix} \bar{q}_2$$

The inverse of the quaternion is found by negating the imaginary component of the quaternion:

$$\bar{q}^{-1} = \begin{bmatrix} -\vec{q} \\ q_4 \end{bmatrix}$$

VI. Performance Analysis

VII. Conclusion

VIII. Future Work

References

- [1] J. Verhage, “Goldman Sachs Has Four Charts Showing the Huge Potential in Virtual and Augmented Reality.” <http://www.bloomberg.com/news/articles/2016-01-13/goldman-sachs-has-four-charts-showing-the-huge-pote> 2016. Accessed: 2016-03-19.
- [2] R. Y. Tsai, “A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses,” *Robotics and Automation, IEEE Journal of*, vol. 3, no. 4, pp. 323–344, 1987.
- [3] J. L. Crassidis, F. L. Markley, and Y. Cheng, “Survey of nonlinear attitude estimation methods,” *Journal of guidance, control, and dynamics*, vol. 30, no. 1, pp. 12–28, 2007.
- [4] E. J. Lefferts, F. L. Markley, and M. D. Shuster, “Kalman filtering for spacecraft attitude estimation,” *Journal of Guidance, Control, and Dynamics*, vol. 5, no. 5, pp. 417–429, 1982.
- [5] M. D. Shuster, “Deterministic three-axis attitude determination,” *Journal of Astronautical Sciences*, vol. 52, no. 3, pp. 405–419, 2004.