Final Progress Report

Lukas Gemar

March 4, 2016

1 Tool Model

The position of the tip of the interaction tool is $\mathbf{x_T^w}$. This position is specified in world coordinates; the superscript w indicates that the coordinates of this position are specified relative to the origin of world space, $\mathbf{O_w}$. This origin of the world coordinate system is approximately the location of the users head. It is not exactly the location of the user's head because the user can look up and down and side to side while wearing the display. The origin of the world coordinate system should remain fixed to these rotations and translations. The rotation of the display is measured, but the translation of the display is not. However, the translation of the head should be a function of the rotation of the display. A simple model of how the head moves from side to side should relate the head's rotation to its movement relative to the neck and shoulders.

Since the camera is mounted to the display, the position of the world coordinate system is not exactly the position of the camera. The origin of the camera coordinate system, $\mathbf{O_c}$, is offset from the world coordinate system by the rotation and translation of the user's head. Assuming that the rotation and translation of the user's head is in fact minimal while the user is seated, then the translation of the camera, $^c\mathbf{t}^w$, should be a function of the rotation of the head mounted display, $^c\mathbf{R}^w$: $^c\mathbf{t}^w = f(^c\mathbf{R}^w)$.

Let $^c\mathbf{R}^w$ and $^c\mathbf{t}^w$ be the rotation and translation matrices, respectively, that specify the transformation from the world coordinate system to the camera coordinate system. Then the position of the interaction tool in camera coordinates, $\mathbf{x_T^c}$, is given by the following relation:

$$\mathbf{x_T^c} = {}^c\mathbf{R^w}\mathbf{x_T^w} + {}^c\mathbf{t^w}$$

The position of the tool relative to the camera, $\mathbf{x_T^c}$, is not observed directly. However, this position is inferred from the position of the back of the tool, $\mathbf{x_E^c}$, and the orientation of the tool. Call the back of the tool the eraser. The position of the eraser, $\mathbf{x_E^c}$, is found through computer vision. The orientation of the tool relative to the world coordinate systsem is specified by the orientation of the tool in body coordinates, $\theta_\mathbf{T}^\mathbf{b}$, and the rotation matrix from

1

the body coordinates of the tool to the world coordinates, $^{w}\mathbf{R}^{b}$; their product yields the absolute orientation of the tool in world coordinates. To correctly display the orientation of the tool to the user, $^{w}\mathbf{R}^{b}$, the rotation of the camera relative to the world, must also be known. $^{w}\mathbf{R}^{b}$ is measured by the virtual reality display. The position of the tool in camera coordiantes is found through the following relation:

$$\mathbf{x_T^c} = \mathbf{x_E^c} + {}^{\mathbf{c}}\mathbf{R^w}({}^{\mathbf{w}}\mathbf{R^b}\theta_\mathbf{T}^\mathbf{b})$$

2 Position Estimation

2.1 Measurement Model

The following is a model of a single frame from the camera. In this frame, the red circular object represents the back of the interaction tool. The back of the interaction tool is called the eraser. The width of the target along the x dimension of the image is d_u.
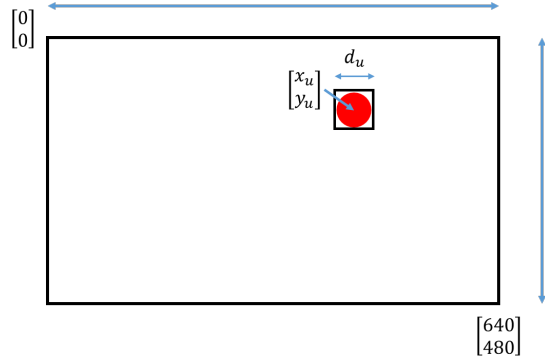
The midpoint of the eraser in the ideal undistorted image is $\mathbf{x_E^u} = \begin{bmatrix} x_u & y_u \end{bmatrix}^\mathbf{T}$. The center of the undistorted image is c_x in the x direction and c_y in the y direction. The focal lengths in the x and y directions are f_x and f_y, respectively. These parameters are found through the camera calibration routine. The measurements of the center of the eraser and its width are related to the hidden state $\mathbf{x_C} = \begin{bmatrix} x_C & y_C & z_C \end{bmatrix}^\mathbf{T}$ 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 [1].

2.2 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 [1]. 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):

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

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

Second, the radial and tangential distortion relations show how the actual image coordinates, $(\mathbf{x_u}, \mathbf{y_u})$, 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:

$$\mathbf{x_u} = \mathbf{x_d}(\mathbf{1} + \mathbf{k_1 r^2} + \mathbf{k_2 r^4} + \mathbf{k_3 r^6})$$

$$\mathbf{y_u} = \mathbf{y_d}(\mathbf{1} + \mathbf{k_1 r^2} + \mathbf{k_2 r^4} + \mathbf{k_3 r^6})$$

and the tangential distortion equation is given by,

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

$$\mathbf{y_u} = \mathbf{y_d} + (\mathbf{p_1}(\mathbf{r^2} + \mathbf{2y^2}) + \mathbf{2p_2 xy})$$

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.

2.3 State Measurements

The observed state consists of the image coordinates of the object and its width in pixels. From the notation derived above, the observed state is represented by the vector $\mathbf{y} = \begin{bmatrix} x_u & y_u & d_u \end{bmatrix}$, where (x_u, y_u) is the location of the center of the object in the undistorted camera image and d_u is its width in pixels.

In order to test the measurement model along each of these degrees of freedom, three manuevers are required: side-to-side tool motion along the x-dimension of the camera coordinate system, up-and-down tool motion along
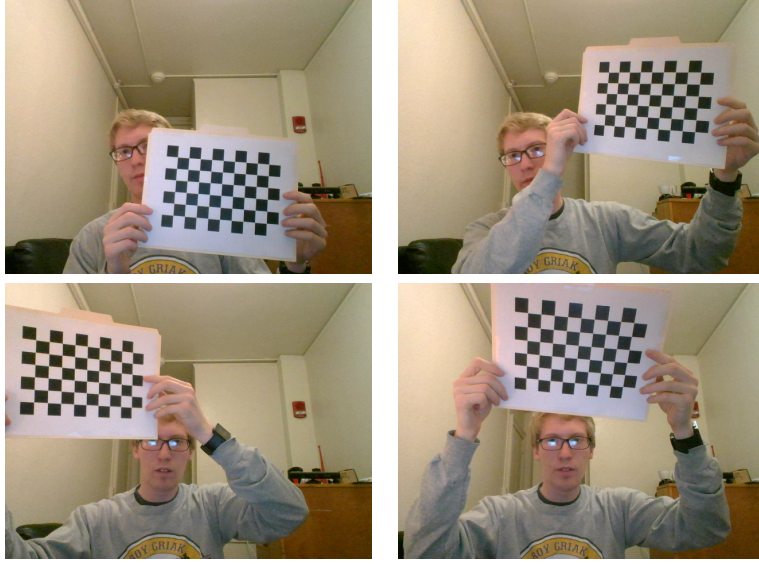
Figure 1: Camera calibration images

the y-axis, and back-and-forth motion towards and away from the camera along the z-dimension. The measurement setup with the camera coordinate system is shown in Figure 2.

The position of the tool eraser in camera coordinates is calculated from the measurements of $\mathbf{y} = \begin{bmatrix} x_u & y_u & d_u \end{bmatrix}$. The raw position observations are shown in Figures , , and ; these are shown side-by-side with the actual tool position in world coordinates.

2.4 Dynamic Model: Constant Velocity

Assume that the acceleration of the interaction tool is small, as is likely to be the case if the interaction tool is moved slowly. If the acceleration is small, then it can be modeled as error in the plant of the dynamic model for an inertial object moving at constant velocity. The dynamic model for an object at constant velocity has state, $\vec{x} = \begin{bmatrix} x & \dot{x} & y & \dot{y} & z & \dot{z} \end{bmatrix}^T$.

$$\dot{\vec{x}}(t) = \mathrm{blkdiag}(A, A, A)\vec{x}(t) + \mathrm{blkdiag}(B, B, B)\vec{w}(t)$$

where

$$A = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \quad \text{and} \quad B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$
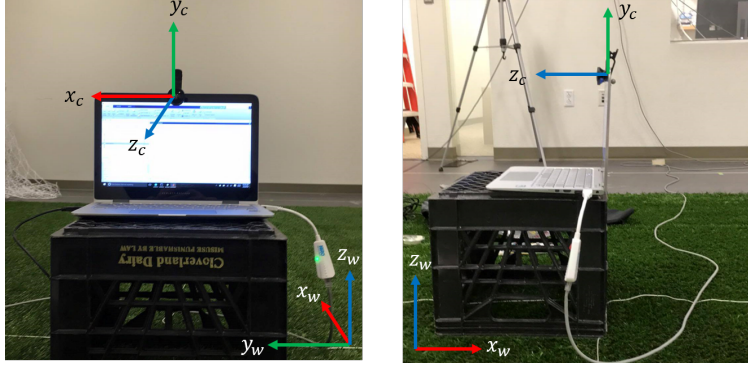
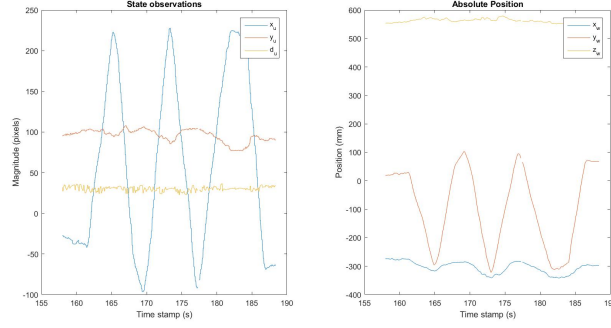Figure 2: Camera coordinate system for measurements

Figure 3: Side to side maneuver

In discrete time, the dynamic model becomes,

$$\vec{x}_{k+1} = \operatorname{diag}(F, F, F)\vec{x}_k + \operatorname{diag}(G, G, G)\vec{w}_k$$

with

$$F = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} \quad \text{and} \quad G = \begin{bmatrix} \frac{T^2}{2} \\ T \end{bmatrix}$$

For the interaction tool, only the position of the tool is measured. Therefore, the observation model is the following:

$$\vec{z}_{k+1} = \operatorname{diag}(H, H, H)\vec{x}_{k+1}$$

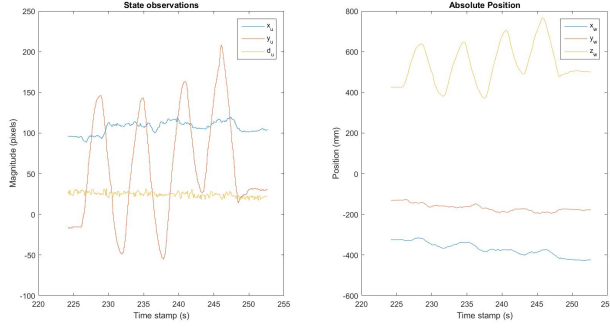with $H = \begin{bmatrix} 1 & 0 \end{bmatrix}$.
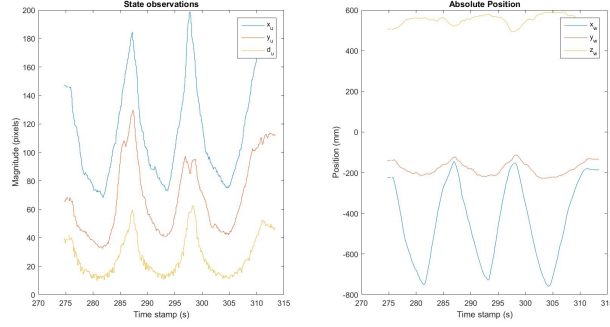
Figure 4: Up and down maneuver

Figure 5: Back to front maneuver

2.5   Hidden State Estimation

3   Attitude Estimation

The attitude 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 the body frame [2]. 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 and angle of rotation. If $\hat{n}$ is the axis of a rotation and θ is the angle of the rotation, then the quaternion $\bar{q}$ represents the attitude:

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

with $\vec{q} = \hat{n}\sin\frac{\theta}{2}$ and $q_4 = \cos\frac{\theta}{2}$. The quaternion must have length 1: $\bar{q}^T\bar{q} = 1$. The attitude matrix and the quaternion representing attitude are related by the following equation, where A is a function of $\bar{q}$ [3]:

$$A(\bar{q}) = (q_4^2 - \|\vec{q}\|^2)I_{3x3} + 2\vec{q}\vec{q}^T - 2q_4[\vec{q}\times]$$

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 $\bar{q}$ can also be expressed as the product of two matrices [2]:

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

with

$$\Xi(\bar{q}) = \begin{bmatrix} q_4\mathbf{I_{3x3}} + [\vec{\mathbf{q}}\times] \\ -\vec{q}^T \end{bmatrix}$$

$$\Psi(\bar{q}) = \begin{bmatrix} q_4\mathbf{I_{3x3}} - [\vec{\mathbf{q}}\times] \\ -\vec{q}^T \end{bmatrix}$$

3.1 Deterministic Attitude Determination

3.1.1 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 $k = 1, 2, 3$ [4]. 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, respecitively, 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,

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

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,

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

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,

$$A = M_b M_r^T$$

3.1.2 TRIAD Estimation Measurements

If A is the rotation matrix from world coordinates to body coordinates, then its inverse, $A^{-1} = A^T$, is the rotation matrix from body coordinates to world coordinates.

A vector $\vec{r}$ in the reference coordinate system is related to a vector $\vec{s}$ in the sensor coordinate system by the relation,

$$\vec{s} = TA\vec{r}$$

where A is the attitude of the interaction tool and T is the rotation matrix that transforms vectors in the body coordinate system to vectors in the sensor coordinate system. T is known as the sensor alignment matrix [4]. The sensor alignment matrix depends on how the sensor is mounted to the rigid body. For the interaction tool, the sensor alignment matrix is,

$$T = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

which is a rotation of 90 degrees clockwise around the z-axis of the body coordinate system. This is the rotation required to align the body coordinate system with the sensor coordinate system, given the mounting position of the inertial measurement unit on the device.

In order to find attitude from measurements of the gravity and b-field vector in the sensor coordinate system, the TRIAD estimation method must be modified to account for the mounting position of the inertial measurement unit. If m_1 is the measured gravity vector in the sensor coordinate system and m_2 is the measured b-field reference vector in sensor coordinates, then $M_s = \begin{bmatrix} s_1 & s_2 & s_3 \end{bmatrix}$, with

$$\vec{s}_1 = \frac{\vec{m}_1}{\|\vec{m}_1\|}, \qquad \vec{s}_2 = \frac{\vec{m}_1 \times \vec{m}_2}{\|\vec{m}_1 \times \vec{m}_2\|}, \qquad \vec{s}_3 = \vec{m}_1 \times \vec{m}_2$$

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

$$M_b = T^{-1}M_s = AM_r$$

Then,

$$A = M_b M_r^{-1} = T^{-1}M_s M_r^{-1}$$

This attitude is the composition of the yaw, pitch, and roll of the interaction tool. Yaw is the rotation of the interaction tool around the z-axis, pitch around the y-axis, and roll around the x-axis.

Yaw maneuver:

Pitch maneuver:

Roll maneuver:

3.2 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
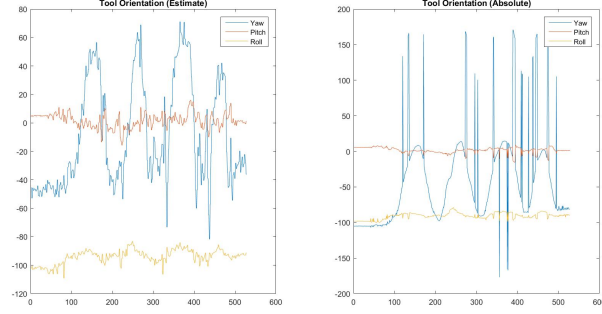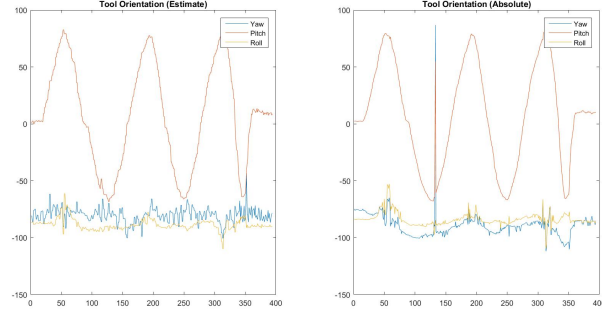
Figure 6: Yaw maneuver

Figure 7: Pitch maneuver

change in the quaternion [3]. 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:

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

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

where the first relation is a state space model with state transition matrix $A(\vec{\omega}) = \Omega(\omega)$, where the state transition matrix is a function of the angular rate vector $\vec{\omega}$:
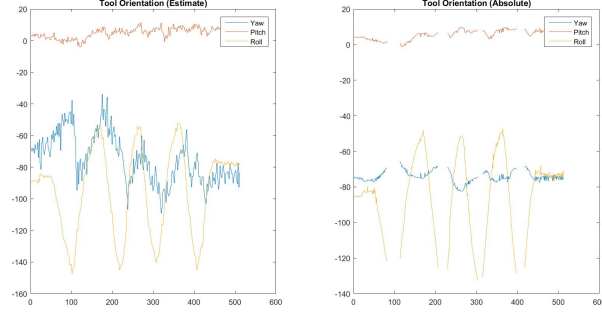
Figure 8: Roll maneuver

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

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 [2]. The following identity holds of the composition of quaternions [3]:

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

with the operation $A(\bar{q})$ finding the attitude matrix A that corresponds with the quaternion $\bar{q}$. Following the notation in [2], 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:

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

11

3.3   Kalman Filter Design

A discrete dynamical system is described by a process model and measurement model. The estimation problem arises in trying to find the hidden state x_k by observing the measurements z_k. In their most general form, the system model and the measurement model are given by the equations,

$$x_{k+1} = Ax_k + Bu_k$$

$$z_k = Hx_k$$

A general estimator or observer is given by the equation [5]:

$$\hat{x}_{k+1} = A\hat{x}_k + L(z_k - H\hat{x}_k) + Bu_k$$

From the dynamic model and the measurement equation, A, B, and H are known. Therefore, it is only necessary to find the observer gain matrix, L.

For the Kalman filter, assume that the the process model and the measurement model are both subject to white noise with zero mean and some covariance. Then the process model and measurement equation become

$$x_{k+1} = Ax_k + Bu_k + G_k w_k$$

$$z_k = Hx_k + v_k$$

$$x_0 \sim (\overline{x}_0, P_{x0})$$

where the assumptions are that w_k and v_k are uncorrelated white noise processes such that $w_k \sim (0, Q_k)$ and $v_k \sim (0, R_k)$.

The Kalman filter has three steps: (1) initialization, (2) time update, and (3) measurement update. During the intialization phase, the state of the model and the covariance are assigned initial values:

$$P_0 = P_{x0} \quad \hat{x}_0 = \overline{x}_0$$

During the time update, also known as the prediction step, the *a priori* estimates for the state and the covariance matrix are given by the following equations:

$$P_{k+1}^- = A_k P_k A_k^T + G_k Q_k G_k^T$$

$$\hat{x}_{k+1}^- = A_k \hat{k} + B_k u_k$$

The measurement update converts the *a priori* predictions for the state and covariance into their *a posteriori* filtered estimates based on the observation of z_k:

$$P_{k+1} = [(P_{k+1}^-)^{-1} + H_{k+1}^T R_{k+1}^{-1} H_{k+1}]^{-1}$$

$$\hat{x}_{k+1} = \hat{x}_{k+1}^{-} + P_{k+1}H_{k+1}^{T}R_{k+1}^{-1}(z_{k+1} - H_{k+1}\hat{x}_{k+1}^{-})$$

Note that during the measurement update, the *a posteriori* estimate is simply the *a priori* estimate plus a correction term. The correction term is simply a gain matrix times the the difference between the actual observation and the predicted observation. The gain matrix is known as the Kalman gain:

$$K_k = P_k H_k^T R_k^{-1}$$

The difference between the actual observation, z_{k+1} and the predicted observation $H_k\hat{x}_{k+1}^{-}$ is known as the residual, ν:

$$\nu = z_{k+1} - H_k\hat{x}_{k+1}^{-}$$

Thus, the measurement update can be written in terms of the *a priori* estimate, the residual, and the Kalman gain:

$$\hat{x}_{k+1} = \hat{x}_{k+1}^{-} + K_k\nu$$

3.3.1 Process Model

Treat the angular rate, $\vec{\omega}$, as the input to the process model. The rate of change of the quaternion is given by the dynamic model,

$$\dot{\bar{q}} = B(\bar{q})\vec{\omega}$$

with the input matrix $B(\bar{q})$:

$$B(\bar{q}) = \frac{1}{2}\Xi(\bar{q}) = \begin{bmatrix} q_4\mathbf{I_{3x3}} + [\vec{\mathbf{q}}\times] \\ -\vec{q}^T \end{bmatrix}$$

Treating the attitude quaternion $\bar{q}$ as roughly constant over a small time interval, the process model can be discretized:

$$\bar{q}_{k+1} = B_k(\bar{q}_k)\vec{\omega}_k$$

3.3.2 Measurement Model

3.3.3 Prediction Equations

The prediction variables are the state of the quaternion at the next time step, $\bar{q}_{k+1}$, and the covariance matrix, $\overline{P}_{k+1}$. The state prediction is computed using the process model:

$$\dot{\bar{q}} = B(\bar{q})\vec{\omega}$$

The covariance prediction is computed

References

[1] Roger 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*, 3(4):323–344, 1987.

[2] John L Crassidis, F Landis Markley, and Yang Cheng. Survey of nonlinear attitude estimation methods. *Journal of guidance, control, and dynamics*, 30(1):12–28, 2007.

[3] Ern J Lefferts, F Landis Markley, and Malcolm D Shuster. Kalman filtering for spacecraft attitude estimation. *Journal of Guidance, Control, and Dynamics*, 5(5):417–429, 1982.

[4] Malcolm D Shuster. Deterministic three-axis attitude determination. *Journal of Astronautical Sciences*, 52(3):405–419, 2004.

[5] Frank L Lewis, Lihua Xie, and Dan Popa. *Optimal and robust estimation: with an introduction to stochastic control theory*, volume 29. CRC press, 2007.