UNIVERSITY OF CALIFORNIA
SANTA CRUZ

VERIFICATION OF TRIAD AND COMPLEMENTARY FILTER ON CUBESAT USING LOW-COST EXTRINSIC VISUAL MARKER TRACKING WITH AN EXTENDED KALMAN FILTER

A thesis submitted in partial satisfaction of the
requirements for the degree of

MASTER OF SCIENCE

in

COMPUTER ENGINEERING

with an emphasis in

ROBOTICS AND CONTROL

by

Jordan Andrew Liss

September 2018

The Thesis of Jordan Andrew Liss
is approved:

Professor Gabriel H. Elkaim, Chair

Professor Renwick E. Curry

Professor Ricardo Sanfelice

Lori Kletzer
Vice Provost and Dean of Graduate Studies

Table of Contents

List of Figures

List of Tables

Abstract

Verification of TRIAD and Complementary Filter on CubeSat using Low-Cost Extrinsic Visual Marker Tracking with an Extended Kalman Filter

by

Jordan Andrew Liss

Very small spacecraft, called SmallSats, CubeSats, and NanoSats, have become very popular due to their low-cost, complexity, and availability of launch platforms (as extra payloads). These small satellites enable a future of low-cost satellite constellations to accomplish various space missions. The Generalized Nanosatellite Avionics Testbed (G-NAT) lab at NASA Ames Research Center in collaboration with UC Santa Cruz has developed a laboratory to verify sensor and actuator performance requirements of CubeSats in a standardized testing facility equipped with an ultra low-cost attitude truth system based on a camera tracking of visual fiducials. This work first developed ultra low-cost real-time streaming attitude truth system that is self-calibrating, portable and scalable. The attitude truth system is cross-platform and can be used at various CubeSat testing facilities to track to CubeSat attitude (angular and translational position in the inertial reference frame). Attitude truth system accuracy and robustness improvements were made to rectify highly variant attitude observations during quick maneuvers using an extended Kalman filter. The attitude truth system, implemented with the extended Kalman filter, was able to sustain estimation standard deviation values of ±1.30°, ±1.30°, and ±0.61° in roll, pitch and yaw, respectively. The finalized truth system with the extended Kalman filter was used as validation for testing and verification of on-board CubeSat attitude estimation. The two attitude estimation algorithms explored, were TRIAD (Three-Axis Attitude Determination) and the complementary filter. Both algorithms were implemented using low-cost IMU sensors and solar photodiodes on a CubeSat testbed. The tests verified that the complementatry filter algorithm is more accurate than TRIAD using the same sensor suite when measured by the attitude truth system; the complementary filter performed better than TRIAD by 3.3°, 3.3° and 2.6° in roll, pitch and yaw, respectively according to the attitude truth system with extended Kalman filter.

To Jeffrey Farrell Liss,

Dad, your fight against Leukemia has made an unshakable impression on me. I wish you were still here.

You have given and will always continue to give me the strength to persevere. I dedicate this thesis to you.

I would also want to dedicate this thesis to Grandma Shirley and Grandpa Sidney.

May you all be guiding us daily through the struggles and triumphs of life.

Acknowledgments

This body of work would not have been possible without the support of many outstanding individuals, institutions and communities. I would first want to thank the NASA Ames Research Center for being my research home away from home.

My sincere appreciation goes to my thesis advisor, Professor Gabriel Elkaim, for giving me the creative research space, weekly feedback and collaborative resources to work with NASA Ames engineers and scientists. Thank you for always being an encyclopedia for sensor fusion, attitude estimation and mechatronic knowledge.

Thank you Ren Curry for always being a great resource of avionics, spacecraft dynamics and controls information. Thank you for always finding time in your busy schedule to discuss my project details.

Also, would like to send my sincere gratitude to Ricardo Sanfelice for your teachings, support and edits.

I would like to thank Matthew Sorgenfrei from the NASA Ames Research Center for finding the time in your daily busy schedule to always discuss my project details with me on the phone or in person.

Lastly, I want to thank the entire UCSC Autonomous Systems Laboratory for the everyday motivation and whiteboard sessions.

Mathematical Notation

$\dot{A}$ = first derivative of A with respect to time.

$\hat{A}$ = estimated value a time-invariant system

$\underline{A}$ = a vector $\in R^n$ where $n \geq 2$.

$[A]$ = a matrix of size $m \times n$ where $m \geq 2$ and $n \geq 2$.

ψ_C^I,θ_C^I,ϕ_C^I = Euler angle of camera reference frame (C) in the inertial frame (I).

ψ_T^C,θ_T^C,ϕ_T^C = Euler angle of tag reference frame (T) in the camera reference frame (C).

ψ_T^I,θ_T^I,ϕ_T^I = Euler angle of tag reference frame (T) in the inertial reference frame (I).

$[R]_C^I$ = rotation matrix from camera reference frame (C)to the inertial frame (I).

$[R]_T^C$ = rotation matrix from the tag reference frame (T) to the camera reference frame (C).

$[R]_T^I$ = rotation matrix from the tag reference frame (T) to the inertial frame (I).

RMS = Root Mean Square = $\sqrt{\frac{\sum_{i=1}^{n} m_i^2}{n}}$

$[\underline{x}]_x$ = skew-symmetric matrix of vector $\underline{x}$

Acronyms

AAS - American Astronomical Society

ADCS - Attitude Determination and Control System

AIAA - American Institute of Aeronautics and Astronautics, Inc.

ASL - Autonomous Systems Laboratory

ATS - AprilTag based attitude truth system

CF - Complementary Filter

COTS - Commercial-off-the-shelf

DCM - Direction Cosine Matrix

DT EKF - Discrete Time Extended Kalman Filter

DOF	Degrees of Freedom
EDSN	Edison Demonstration of SmallSat Networks
ESD	Electro-static discharge
EKF	Extended Kalman Filter
G-NAT	Generalized Nanosatellite Avionics Testbed
LEO	Lower Earth Orbit
NASA	National Aeronautics and Space Administration
OpenCV	Open source computer vision libraries
SLAM	Simultaneous Localization And Mapping
SD	Standard Deviation
TRIAD	Three-Axis Attitude Determination
UAV	Unmanned Aerial Vehicles
UCSC	University of California Santa Cruz

Chapter 1

Background

1.1 Introduction

Very small spacecraft, called SmallSats, CubeSats, and NanoSats, have become very popular due to their low-cost, complexity, and availability of launch platforms (as extra payloads). Small satellite constellations are being tested and operated to create new scientific experiments in low Earth orbit (LEO) and beyond. For instance, a constellation of 50 or 100 miniature satellites have the potential to monitor space weather with such high resolution (15 minute resolution) that researchers could detect the various microclimates around the globe. Obtaining high quality sensor data at different spatial and temporal domains simultaneously is invaluable for not only the weather applications but for military, space, transportation, and communication applications.[3]

For the purpose of this work a CubeSat is defined as any spacecraft that adheres to the CubeSat Standard, a 10 cm x 10 cm x 10 cm cube comprises one unit of volume

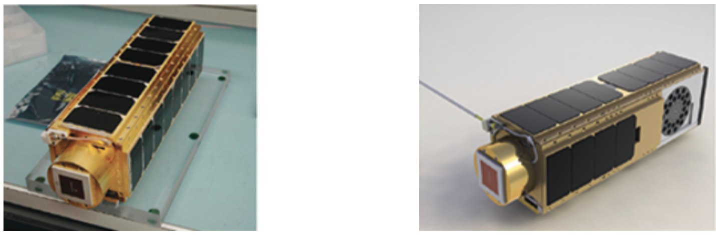


Figure 1.1: PharmSat(Left) and OREOS(Right)

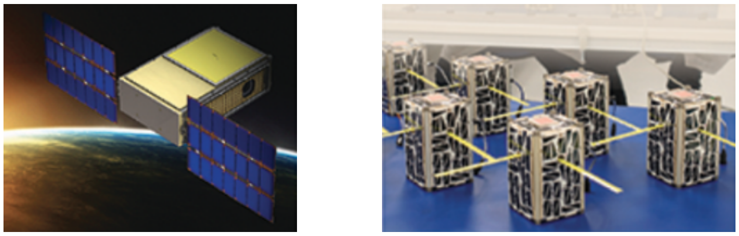


Figure 1.2: Biosentennial(Left) and EDSN(Right)

(abbreviated as 1U). The majority of CubeSats that have been launched to date have ranged in size from 1U to 3U, with an overall volume of no more than 10 cm x 10 cm x 30 cm. NASA Ames has a rich history of building 3U CubeSats to research topics in fundamental space biology, such as the Biosentennial, O/OREOS, and PharmaSat spacecraft (Fig. 1.1) [4]. More recently, NASA Ames developed a constellation of eight 1.5U CubeSats for the Edison Demonstration of SmallSat Networks (EDSN) mission (Fig. 1.2). The EDSN mission which would have demonstrated multi-point science operations in LEO. [5] Unfortunately, EDSN suffered a launch vehicle failure prior to deployment.

The primary reason that smallsats have gained in popularity is primarily one of cost; the overall cost of a large complicated satellite can be distributed over a set of smaller satellites each with their own individual sensors, communications, and computer systems. The creative freedom provided by the boom in smallsat projects has created a need for flexible, easily accessible facilities for iterative hardware-in-the-loop (HIL) and software-in-the-loop (SIL) testing for performance validation of CubeSat attitude determination and control system (ADCS) technologies. Generally the validation and iterative testing requirements for real-time satellite attitude estimation can accrue large costs (especially for clustered systems) due to time lost setting up a unique test environment for each mission and from the lack of standardized measurements to fine tune sensor integration and engineering test systems.

The Generalized Nanosatellite Avionics Testbed (G-NAT) lab at NASA Ames Research Center in collaboration with UC Santa Cruz has developed a laboratory to verify sensor and actuator performance requirements of CubeSats in a standardized testing facility. The objective of this lab is to advance the technology readiness level of ADCS and avionics technologies for small spacecraft missions under development by NASA and its partners in academia and industry. The focal point of the lab is a pair of test facilities for studying rotational motion operations, each of which is equipped with a three degree-of-freedom (DOF) rotational air bearing, a Helmholtz cage for generating a time-varying magnetic field, and sun emulators. For this research one of the two test facilities was augmented with the COTS web cam, and motion data was collected during a series of rotational maneuvers performed on the rotational air bearing.

This thesis is organized as follows: Chapter 1 describes the background, motivation and test environment for the CubeSat facility, CubeSat dynamics and mathematical background for attitude estimation (using Direction Cosine Matrices) and attitude propagation modeled as a discrete time linear dynamical system. The first chapter additionally provides background for the motivation for a turnkey ”attitude truth” system (ATS) using inexpensive cameras to complement the G-NAT facility.

Chapter 2 describes the product research for an accurate inexpensive ATS, extrinsic calibration scheme development and tracking improvements using a discrete Kalman filter (DKF). The chapter details the design of the ATS system to be ultra-cheap, scalable, portable, accurate and easy to implement for CubeSat ADCS testbed validation.

Chapter 3 describes the implementation of sensor calibration and validation of two attitude estimation algorithms tested on-board the Cubesat ADCS testbed. A sun sensor from a photodiode tetrahedron is used to determine a tri-axial vector of the sun direction using a least-squares estimation. The on-board magnetometer is calibrated using a iterative least-squares calibration method. Additionally, a batch mis-alignment reorientation calibration is performed on the two Cubesat ADCS testbed sensors to have their reference frames coincident. After sensor calibration the tri-axial vectors are the inputs for attitude estimation tests. Two attitude estimation methods the three-axis attitude determination (TRIAD) and the explicit complementary filter (ExCF) are tested on-board the ADCS testbed and results were validated by the ATS.

The final chapter describes a summary of the G-NAT work and future work that can be performed to improve the ATS, the G-NAT facility, ADCS testbed and essential infrastructure for swarm satellite constellations.

1.2 Motivation

A wide range of science experiments executed by a swarm of spacecraft in LEO require station-keeping on the part of the members of the swarm, which is often accomplished using propulsion systems. However, the strict mass and volume restrictions levied on CubeSats make manifesting propulsion capabilities challenging. Consider instead the situation depicted in Figure 1.3, in which a CubeSat uses differential drag to alter its orbital velocity. As seen by performing a single-axis rotation maneuver the CubeSat can alter its surface area (Change C_f) normal to the velocity direction, which will in turn change its aerodynamic drag (Described in equation 1.1).

$$F_d = \frac{1}{2} C_f \rho v^2 \tag{1.1}$$

Where ρ=air density, C_f =drag coefficient dependent on surface area normal to velocity direction, v = object velocity

Thus, members of a swarm could use attitude control maneuvers to alter their orbital velocity, allowing for control of the relative distances between the members. [6] For swarm operations, it would be important for each spacecraft to be able to communicate their current attitude state both to each other and to the ground station in order to inform future orbit maintenance maneuvers. This operational scenario is currently being studied in the G-NAT lab at NASA Ames in support of multiple future LEO missions.

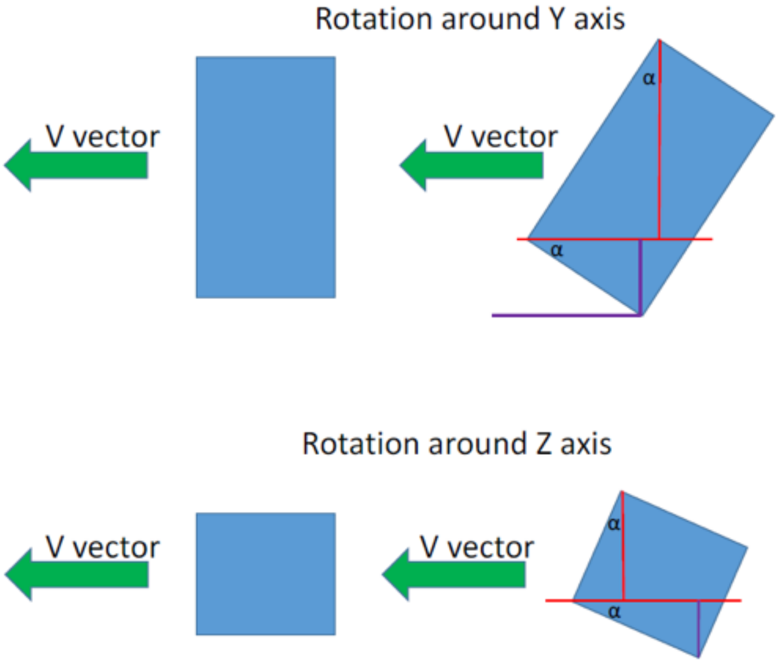


Figure 1.3: A representative example of how a CubeSat could perform a single-axis reorientation to change its surface area with respect to the velocity (v) direction

1.2.1 Generalized Nanosatellite Avionics Testbed Laboratory Facility

The G-NAT lab is a collaboration between the Mission Design Division, the Intelligent Systems Division, and the Engineering Systems Division at NASA Ames Research Center, the goal of which is to advance the development of hardware and software for actively controlled CubeSats. As can be seen in Fig. 1.4, the focal point of this lab is a pair of test facilities that enable full state determination when using a sensor combination that is typical for CubeSats operating in low Earth orbit (LEO). Each test facility uses an air bearing to allow for 3-DOF rotational motion (With roll and pitch limits of $\pm 10^o$); one of these air bearings is designed for CubeSats measuring 1U to 3U in size, and the second air bearing can support CubeSats measuring anywhere

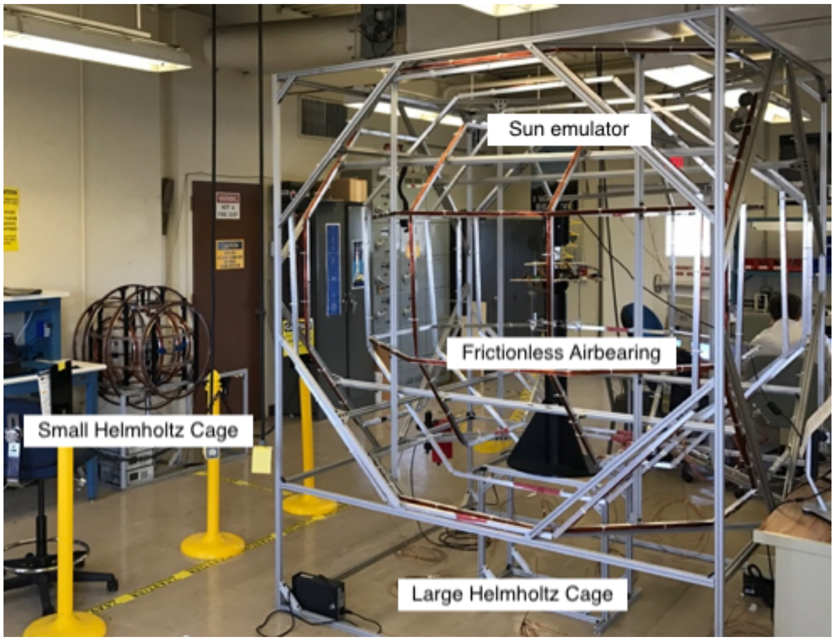


Figure 1.4: The G-NAT lab at NASA Ames Research Center consists of an air bearing to allow for 3-DOF rotational motion surrounded by a Helmholtz cage connected to a power supply generating a time-varying magnetic field.

from 3U to 12U in size. Surrounding each of the air bearings is a Helmholtz cage, a test apparatus capable of generating a time-varying magnetic field. This magnetic field is driven by a set of programmable switching power supplies that communicate with the ground station computer associated with each test facility. An orbit propagator running in MATLAB calculates what the magnetic field should be at the orbit of choice for the particular test program and then sends corresponding current commands to the power supplies. Finally, each test facility also uses a COTS sun emulator bulb to simulate the solar vector which would be measured in LEO, which has been explored by many other CubeSat architecture for its simplicity and low-cost sensor approach to accurate attitude determination. [2],[1] Thus, by measuring the magnetic field vector, the sun vector, and its body-fixed angular rates, a testbed can determine its attitude and angular velocity state with respect to the inertial frame of the lab.

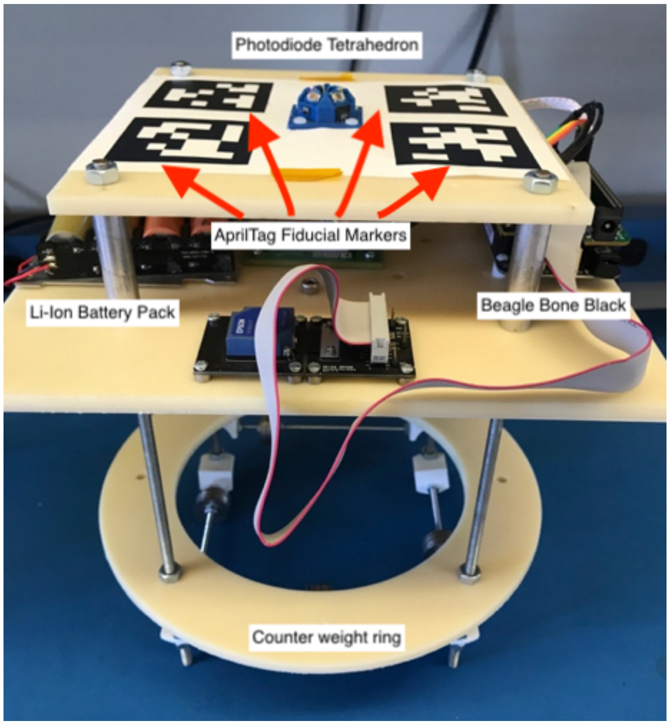


Figure 1.5: The attitude determination electro-static discharage (ESD) approved testbed equipped with board-level computer (ie. Beagle Bone Black), tetrahedron of sun sensors, low-cost IMU sensors and AprilTag object detection stickers

The ADCS testbed used for this particular ADCS research in the G-NAT lab can be seen in Fig. 1.5. A tetrahedron of sun sensors is mounted on the top deck of the testbed for measuring the sun vector, and a MEMS inertial measurement unit (IMU), seen on the bottom deck, can measure both testbed angular rates and the magnetic field generated by the Helmholtz cage. The ring of plastic at the very bottom of the testbed is a hard stop for the testbed in the event that it becomes too imbalanced, and also serves as a mechanism by which to add dummy mass to lower the center of gravity of the testbed. The ring of plastic limits the testbed roll and pitch range of motion to $\pm 10^o$ and leaves no limits on range of motion for yaw. This additional mass is obviously not flight-like, but it greatly improves the stability of the testbed during operations. One drawback to the lower center of gravity is that the testbed is more susceptible to cross-axis rotational motion, which can sometimes make verification of space-based attitude control operations challenging. The yellowish hue of the plastic is the result of a doping agent used to prevent the plastic from storing charge, as is typical of an insulator. Many of the components tested in the G-NAT lab are sensitive to electro-static discharge (ESD), so this is a necessary precaution.

The testbed is controlled by a Beaglebone Black single-board computer, which runs the Debian Linux open source operating system. Use of the Linux operating system is an important feature of the testbed, since many NASA Ames flight programs make use of the Core Flight Executive/Core Flight Software (cFE/cFS) architecture, a software executive which can run on Linux or certain real-time operating systems. Software drivers for the various sensors and actuators are written as software applications, which communicate with each other via cFE/cFS. These sensor and actuator I/O applications interact with attitude determination and attitude control applications, which are auto-coded from MATLAB (Or C/C++) using a partially customized tool-chain developed at NASA Ames.

An obvious drawback in the current layout of the G-NAT lab is that there is no direct support for hardware-in-the-loop (HIL) testing of star trackers. An increasing number of CubeSat-class missions, such as those planned for deep space operations, require a star tracker for accurate attitude determination. However, lab testing of star trackers can be very challenging from both a hardware and a software perspective. [1] Unless a simulated star field can be mounted directly to the star tracker being tested, it is necessary to create some form of external star field, which often requires modifying software internal to the tracker to account for the difference between the 2D star field and 3D space. These software modifications hamper the ”flight like” nature of the testing, and as such this is not an approach considered in the G-NAT lab. However, there are still many examples of CubeSat missions which accomplished their attitude determination goals without the use of a star tracker and these solutions are the current focus of the lab.

The G-NAT lab verification of ADCS performance is the turnkey ”attitude truth” system (ATS) using inexpensive cameras. A variety of motion capture systems have been used in the past for verification of ADCS performance, but implementing these systems can be prohibitively expensive when compared to the development costs of many typical CubeSat missions (At university and government institutions). [7]

1.3 Euler Equations and Direction Cosine Matrix

An object’s attitude can be thought of as its 3D orientation in space, and it is usually defined by relating an object’s local coordinate frame to a global reference frame. The local frame is often called the “body” frame, while the global reference is usually referred to as the “earth” or “inertial” frame. The relationship between the body frame and the earth frame can be expressed using many different methods and nomenclatures. Each of these methods offers a unique way to interpret the attitude of an object. For a detailed discussion on this topic see the work of Shuster [8]. This section will provide a brief introduction to two common forms of attitude parameterization that will be used throughout this work.

1.3.1 Euler Angles

Euler angles parameterize attitude by using a length 3 vector. This vector will be denoted by

$$EulerAngles = \begin{bmatrix} \psi \\ \theta \\ \phi \end{bmatrix} \tag{1.2}$$

Each element of this vector is a scalar value representing a rotation angle about one of the body axes and together, the angles ψ, θ, and ϕ, are known as Euler angles.

The Euler angle formulation chosen for this thesis is the 3-2-1 Euler set, and is commonly used in aviation.

When the above convention is used, the angles ϕ, θ, and ψ are given the names roll, pitch and yaw, respectively. These rotations can be visualized by Fig. 1.6. A key advantage to the use of Euler angles is their intuitive nature. An engineer can very easily inspect the Euler angles for a given object and immediately inference an understanding of its orientation. Also, the use of Euler angles can represent an object's attitude with only three numbers. It will be seen later that other methods require more parameters.[9]

The biggest disadvantage to the use of Euler angles is the existence of a mathematical singularity at a pitch angle of $\pm\pi/2$. This singularity can cause algorithms using Euler angles to fail (a phenomenon known as gimbal lock).[9] The kinematic Eu-

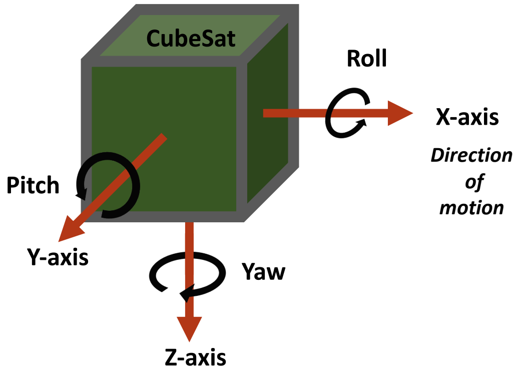


Figure 1.6: As you can see x-axis is out the nose where roll is counter-clockwise rotation about the x-axis. Y-axis is out the right wing and pitch is counter-clockwise rotation about the y-axis and z is down towards the Earth's center. These angles parameterize attitude by using a length 3 vector.

ler angle velocity vector similar to reference [10] is described in equation 1.3 and issues of these singularities can be investigated more clearly. The coordinate equation is:

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & \sin(\phi)\tan(\theta) & \cos(\phi)\tan(\theta) \\ 0 & \cos(\phi) & -\sin(\phi) \\ 0 & \sin(\phi)/\cos(\theta) & \cos(\phi)/\cos(\theta) \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{1.3}$$

The singularities at $\theta = \pm\pi/2$ will cause numerical problems. When $\theta = \pm\pi/2$ $\dot{\psi}$ can go to infinity. The on-board attitude estimation algorithms using testbed sensors take Euler angles and converts into rotation matrices or direction cosine matrices (DCM) to rotate an arbitrary vector in the inertial reference frame. The rotation matrix or DCM can then be multiplied by the earth-frame or inertial reference-frame vector without propagating the singularities issues just described.

1.3.2 Direction Cosine Matrix (DCM)

The direction cosine matrix (DCM) parameterizes attitude by the use of a 3 by 3 orthonormal rotation matrix. This matrix relates the basis vectors of the earth coordinate frame to the basis vectors of the body frame. Specifically, each (i, j) element of the DCM is the cosine of the angle between the i^{th} basis of the global frame to the j^{th} basis of the local frame. [11] In the context of Euler angles, the DCM can be written as:

The rotation matrix describing rotating about the x-axis:

$$R^x = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\phi) & \sin(\phi) \\ 0 & -\sin(\phi) & \cos(\phi) \end{bmatrix} \tag{1.4}$$

The rotation matrix describing rotating about the y-axis:

$$R^y = \begin{bmatrix} \cos(\theta) & 0 & -\sin(\theta) \\ 0 & 1 & 0 \\ \sin(\theta) & 0 & \cos(\theta) \end{bmatrix} \tag{1.5}$$

The rotation matrix describing rotating about the z-axis:

$$R^z = \begin{bmatrix} \cos(\psi) & \sin(\psi) & 0 \\ -\sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{1.6}$$

Multiple all the individual direction cosine matrices for each axis rotation to obtain the total 3 by 3 DCM matrix represented by R

$$[R] = [R^x][R^y][R^z] \tag{1.7}$$

$$[R] = \begin{bmatrix} \cos\psi\cos\theta & \sin\psi\cos\theta & -\sin\theta \\ \cos\psi\sin\theta\sin\phi - \sin\psi\cos\phi & \sin\psi\sin\theta\sin\phi + \cos\psi\cos\phi & \cos\theta\sin\phi \\ \cos\psi\sin\theta\cos\phi + \sin\psi\sin\phi & \sin\psi\sin\theta\cos\phi - \cos\psi\sin\phi & \cos\theta\cos\phi \end{bmatrix} \tag{1.8}$$

The DCM is orthonormal and can be used directly to rotate arbitrary vectors from one coordinate frame to another. Moreover, propagating a DCM from one time step to another is a computationally simple integration (e.g. when using the matrix exponential). This is a distinct advantage over the use of Euler angles. Another advantage is that the DCM does not have any mathematical singularities. This makes it an appealing choice when avoiding gimbal lock. Obtaining Euler Angles from DCM and visa versa is used extensively in this work to provide intuitive information about attitude of the ADCS for the facility operator.

The process to go from DCM back to Euler angles can be determined using equations 3.15 -3.17 from the reference [9]

$$\phi = atan(R_{23}/R_{33}) \tag{1.9}$$

$$\theta = asin(-R_{13}) \tag{1.10}$$

$$\psi = atan(R_{12}/R_{11}) \tag{1.11}$$

1.4 Non-Linear Model and Extended Kalman Filter (EKF)

The work performed in the G-NAT lab on the fiducial tag based tracking system leverages rigid body motion and Euler angle differential equations of a 3 DOF model seen in reference [9] as illustrated in Fig. 1.6. The model will be used to estimate Euler angles, Euler angular rates and body rates of the testbed from camera data. The attitude truth system assumes the reference frame of the testbed is coincident with the inertial reference frame (located on the top center of the frictionless air bearing tower). Euler angles are denoted as ϕ, θ and ψ, respectively and body rates of the testbed are denoted as p, q and r, respectively.

As seen previously the dynamics of Euler angles is described by equation 1.3,

$$\begin{bmatrix}\dot{\phi}\\ \dot{\theta}\\ \dot{\psi}\end{bmatrix} = \begin{bmatrix}1 & \sin(\phi)\tan(\theta) & \cos(\phi)\tan(\theta)\\ 0 & \cos(\phi) & -\sin(\phi)\\ 0 & \sin(\phi)/\cos(\theta) & \cos(\phi)/\cos(\theta)\end{bmatrix}\begin{bmatrix}p\\ q\\ r\end{bmatrix} \tag{1.12}$$

and the dynamics of the angular rates is described by the rigid body motion equation as seen in reference [12]

$$J\dot{\omega} = (J\omega)_x\omega + \tau \tag{1.13}$$

where J is the moment of inertia tensor of the CubeSat testbed assuming a constant uniform density cube, $\dot{\omega} = [\dot{p}, \dot{q}, \dot{r}]$ is the rate of the body angular rates, τ is an external control torque, $(J\omega)_x$ is a skew symmetric matrix. For this study the external control torque is zero with some noise. The constant uniform density moment of inertia tensor of the CubeSat testbed for this research is

$$J = \begin{bmatrix}J_{xx} & 0 & 0\\ 0 & J_{yy} & 0\\ 0 & 0 & J_{zz}\end{bmatrix} \tag{1.14}$$

By multiplying both sides of equation 1.13 by the inverse of J the rigid body motion equation can be used to solve for the rates of the body angular rates

$$\begin{bmatrix}\dot{p}\\ \dot{q}\\ \dot{r}\end{bmatrix} = \begin{bmatrix}\frac{1}{J_{xx}}(-J_{zz}rq + J_{yy}qr)\\ \frac{1}{J_{yy}}(J_{zz}rp - J_{xx}pr)\\ \frac{1}{J_{zz}}(-J_{yy}qp + J_{xx}pq)\end{bmatrix} \tag{1.15}$$

The states of the system are as follows,

$$\underline{x} = \begin{bmatrix}x_1\\ x_2\\ x_3\\ x_4\\ x_5\\ x_6\end{bmatrix} = \begin{bmatrix}\phi\\ \theta\\ \psi\\ p\\ q\\ r\end{bmatrix} \tag{1.16}$$

In continuous form the CubeSat testbed model is represented by the state space form,

$$\underline{\dot{x}} = \underline{f}(\underline{x}) = \begin{bmatrix} (p_k + \sin(\phi_k)\tan(\theta_k)q_k + \cos(\phi_k)\tan(\theta_k)r_k) \\ (\cos(\phi)q_k - \sin(\phi)r_k) \\ (\frac{\sin(\phi_k)}{\cos(\theta_k)}q_k + \frac{\cos(\phi_k)}{\cos(\theta_k)}r_k) \\ \frac{1}{J_{xx}}(-J_{zz}r_kq_k + J_{yy}q_kr_k) \\ \frac{1}{J_{yy}}(J_{zz}r_kp_k - J_{xx}p_kr_k) \\ \frac{1}{J_{zz}}(-J_{yy}q_kp_k + J_{xx}p_kq_k) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ \xi_p \\ \xi_q \\ \xi_r \end{bmatrix} \tag{1.17}$$

where $\underline{f}$ is a non-linear function of the states. The variables p, q and r are unknown to the measurement system and must be modeled to include some zero-mean white noise ξ_p, ξ_q, and ξ_r, respectively.

The full CubeSat testbed non-linear dynamical model in implementation for this study is converted into discrete time state space form based on the equations from reference [13]

$$x_{k+1} = \Phi_k x_k + B_k u_k + \Gamma\underline{\xi} \tag{1.18}$$

$$\tag{1.19}$$

where $\underline{x}_k$ are the states and has the dimensions $n \times 1$, Φ_k is the transition matrix and has the dimensions $n \times n$, B_k is the input matrix and has the dimensions $n \times m$, Γ is the process noise distribution, and $\underline{\xi}$ zero mean Gaussian noise. Each iteration is denoted by the index, k.

We shall investigate the problem of estimating $\underline{x}_k$ from the sampled discrete measurements of the form

$$y_{k+1} = H_k x_k + D_k u_k + \underline{v} \tag{1.20}$$

H_k is the observation matrix and has the dimensions $p \times n$, D_k is the feedthrough matrix and has the dimensions $p \times m$, and v_k is white random sequence of zero mean Gaussian random variables. [13]

The control inputs for the model is assumed to be a zero vector, because the CubeSat testbed is not equipped with an actuation system.

$$\underline{u}(t) = \left[0_{[6x1]}\right] \tag{1.21}$$

The state transition matrix Φ_k of the system in state space form is

$$\Phi_k = e^{A\Delta t} \tag{1.22}$$

where Δt is the time between each sample and

$$A = \begin{bmatrix} \Phi_{1,1} & \Phi_{1,2} & 0 & 1 & \sin(\phi)\tan(\theta) & \cos(\phi)\tan(\theta) \\ \Phi_{2,1} & 1 & 0 & 0 & \cos(\phi) & -\sin(\phi) \\ \Phi_{3,1} & \Phi_{3,2} & 1 & 0 & \frac{\sin(\phi)}{\cos(\theta)} & \frac{\cos(\phi)}{\cos(\theta)} \\ 0 & 0 & 0 & 1 & \Phi_{4,5} & \Phi_{4,6} \\ 0 & 0 & 0 & \Phi_{5,4} & 1 & \Phi_{5,6} \\ 0 & 0 & 0 & \Phi_{6,4} & \Phi_{6,5} & 1 \end{bmatrix} \tag{1.23}$$

and

$$\Phi_{1,1} = 1 + (\cos(\phi)\tan(\theta)q - \sin(\phi)\tan(\theta)r) \tag{1.24}$$

$$\Phi_{1,2} = (\sec^2(\theta)\sin(\phi)q + \sec^2(\theta)\cos(\phi)r) \tag{1.25}$$

$$\Phi_{2,1} = (-\sin(\phi)q - \cos(\phi)r) \tag{1.26}$$

$$\Phi_{3,1} = (\frac{\cos(\phi)}{\cos(\theta)}q - \frac{\sin(\phi)}{\cos(\theta)}r) \tag{1.27}$$

$$\Phi_{3,2} = ((\sin(\phi)\cos^{-2}(\theta)\sin(\theta))q + (\cos(\phi)\cos^{-2}(\theta)\sin(\theta))r) \tag{1.28}$$

$$\Phi_{4,5} = \frac{1}{J_{xx}}(-J_{zz}r + J_{yy}r) \tag{1.29}$$

$$\Phi_{4,6} = \frac{1}{J_{xx}}(-J_{zz}q + J_{yy}q) \tag{1.30}$$

$$\Phi_{5,4} = \frac{1}{J_{yy}}(J_{zz}r - J_{xx}r) \tag{1.31}$$

$$\Phi_{5,6} = \frac{1}{J_{yy}}(J_{zz}p - J_{xx}p) \tag{1.32}$$

$$\Phi_{6,4} = \frac{1}{J_{zz}}(-J_{yy}q + J_{xx}q) \tag{1.33}$$

$$\Phi_{6,5} = \frac{1}{J_{zz}}(-J_{yy}p + J_{xx}p) \tag{1.34}$$

In order to be robust and flexible, the vision system is not dependent on external knowledge from the testbed sensors. The on-board sensors of the ADCS testbed do not exchange information with the vision system. The initial angular velocity guess (to initialize the state estimation of the EKF) is thus approximated from the first two angular positions data sets from the attitude truth system.

$$p(1) = \frac{\phi(2) - \phi(1)}{\Delta t} \tag{1.35}$$

$$q(1) = \frac{\theta(2) - \theta(1)}{\Delta t} \tag{1.36}$$

$$r(1) = \frac{\psi(2) - \psi(1)}{\Delta t} \tag{1.37}$$

The measurement model is provided by the attitude truth system measurements, which are absolute Euler angles with respect to the inertial reference frame and assumed to include zero-mean Gaussian-distributed noise $v_\theta \sim N(0, V_\theta)$, $v_\phi \sim N(0, V_\phi)$ and $v_\psi \sim N(0, V_\psi)$.

$$\underline{y}_k = \begin{bmatrix} \phi_m(k) \\ \theta_m(k) \\ \psi_m(k) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \underline{x}_k + \begin{bmatrix} v_{\phi_m} \\ v_{\theta_m} \\ v_{\psi_m} \end{bmatrix} \tag{1.38}$$

Using the non-linear rigid body motion equations, Euler angle differential equations and discrete measurements model a state space configuration can be used by the extended Kalman filter to smooth noisy AprilTag data measurements. The attitude truth system with extended Kalman filter improves functionality of the attitude tracking system to be within the G-NAT facility accuracy goals for a majority of CubeSat missions.

Actual CubeSat motion obtained data is taken from a web-cam for this study and the EKF implementation is discussed in chapter 3.

Chapter 2

Camera Attitude Truth System (ATS)

2.1 Introduction

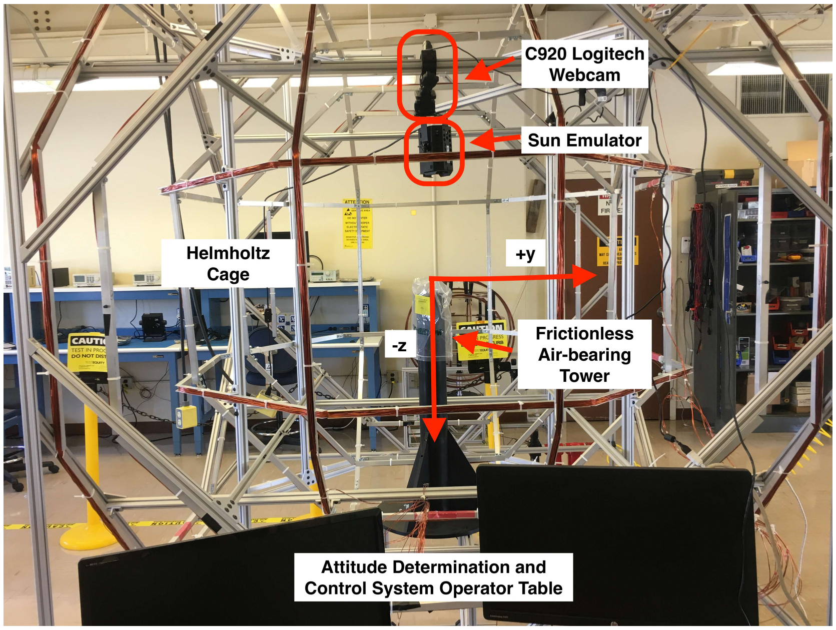

C920 Logitech Webcam
Sun Emulator
Helmholtz Cage
+y
-z
Frictionless Air-bearing Tower
Attitude Determination and Control System Operator Table

Figure 2.1: G-NAT facility from point of view of the operator with the inertial reference frame, $+y$ to the right and $+x$ towards the operators Point of View (POV)

A variety of motion capture systems have been used in the past for verification of ADCS performance, but implementing these systems can be prohibitively expensive when compared to the development costs of many typical CubeSat missions. When deciding which low-cost tracking system to use for ADCS testing validation in the G-NAT lab see Fig. 2.1. Several requirements were identified as critical to the lab mission:

1) Accurate to within standard deviation(SD) of ±2.5°
2) A live stream with runtime object attitude read-outs for the test operator to see in real-time.
3) A high fidelity data acquisition system with a rate of at least 10 hz.
4) A seamless software environment that is portable and scalable.
5) Text file saving functionality and on-demand video and photo saving functionality.
6) A simple and easy way to implement a extrinsic calibration scheme that aligns the camera frame to the inertial frame of the facility.
7) A web-based interface to the G-NAT lab test environment to publish data for remote collaborators.

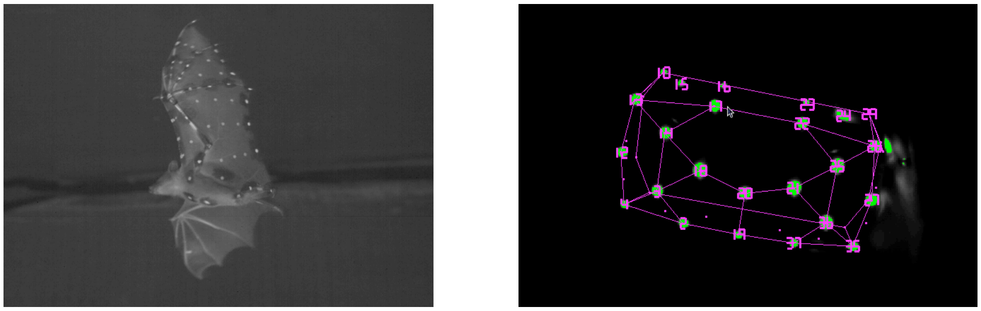
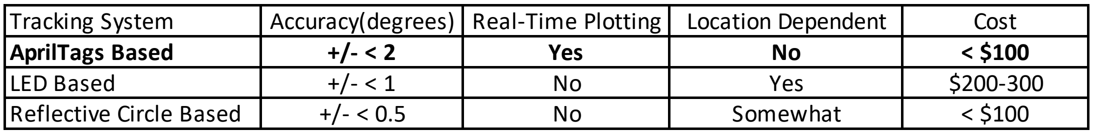


Figure 2.2: Reflector based tracking system following flight of a bat(left) and LED based tracking system tracking attitude of a small Oculus Rift Headset(right)

Prior to converging onto the AprilTag based attitude truth system, two other tracking systems were explored, including a reflector based and LED based tracking system as seen in Fig. 2.2. The reflector object based attitude truth system is by far a more accurate version of a low-cost tracking system (millimeter accuracy), however it requires human input to identify pixel subsections in the succession of image frames to track. [14],[15] The succession of image subsections are compared using a the cross-correlation algorithm and a variation of the Direct Linear Transform algorithm to determine the projection matrix of the reflector marker in the global reference frame from the 2D image. [16] The algorithm is very prone to losing the reflector marker centroid in a succession of image frames in the G-NAT lab test environment. Because the identified pixel subsection depends heavily on consistent lighting and high frame rate which is difficult to implement in the G-NAT lab when using a sun emulator. The LED based tracking system has almost the same attitude accuracy levels as AprilTag, but requires specific configurations, wiring, an energy source and again some consistent lighting. Additionally, both systems may be more accurate, but both are designed to be analyzed in post-processing. When performing iterative testing both these systems are not as ideal as an AprilTag based external reference as presented in the table figure 2.1.



Tracking System	Accuracy(degrees)	Real-Time Plotting	Location Dependent	Cost
AprilTags Based	**+/- < 2**	**Yes**	**No**	**< $100**
LED Based	+/- < 1	No	Yes	$200-300
Reflective Circle Based	+/- < 0.5	No	Somewhat	< $100

Table 2.1: Comparison Chart of existing low-cost attitude tracking systems.

The AprilTag attitude truth system may not be as accurate as the competing methods, but as seen in the table 2.1 it is ideal for CubeSat tests in the G-NAT lab, because of the seamless implementation, robustness to lighting fluctuations, ultra-low cost, low maintenance, and real-time plotting capabilities. The cost metric, includes cost of web-cam and visual marker materials (i.e. the LED system includes the batteries). The eventual chosen approach to verification of CubeSat-class ADCS technologies in the G-NAT at NASA Ames Research Center is to use the open source AprilTag software package with a COTS high-definition web cam. The AprilTag software uses the web cam to track the motion of visual targets mounted on a test article and reports back roll, pitch, yaw, and Cartesian position information in real time. The visual targets can be printed on any type of standard paper, and a wide variety of COTS cameras have been shown to yield valid measurements. [17], [18]

Section 2.2 will explain briefly the AprilTag OpenCV project and its approach to computer vision tracking which makes use of 2D bar code style tags to enable 6-DOF localization of features from a single image.

Section 2.3 will explain the AprilTag based attitude truth system extrinsic calibration scheme to determine the transformation from the tag camera reference frame $[R]_T^C$ into the inertial reference frame $[R]_T^I$. The attitude truth system extrinsic calibration uses an external ”truth” plate to calculate the rotation matrix of the camera in the inertial reference frame.

Finally, section 2.4 will explain the full-scale integrated implementation of attitude truth system which takes the calibrated AprilTag tags extrinsically calibrated into the inertial reference frame and uses a extended Kalman filter to make the attitude truth system attitude output more accurate and robust during testbed position attitude maneuvers.

2.2 OpenCV AprilTag

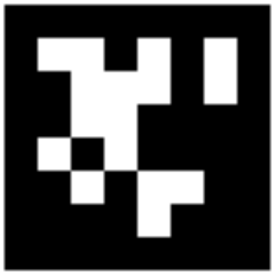


Figure 2.3: AprilTag 2D bar code style ”tags” to enable 6-DOF localization of features from a single image.

The AprilTag system is an approach to computer vision tracking which makes use of 2D bar code style tags to enable 6-DOF localization of features from a single image as seen in Fig. 2.3. While a number of localization schemes using LEDs or reflector circles are reported in the literature, a unique feature of the AprilTag based ATS approach is that the fiducials can be cheaply printed on computer paper, require no ancillary wiring or power, and can be placed virtually anywhere on the test article. Beyond the simplicity of the implementation and the open source nature of the software, the major advantages of the AprilTag base approach are the robustness of the detection algorithm and the localization accuracy. Specifically, a graph-based image segmentation algorithm is employed, which allows for precise estimation of the location of the tags even in a noisy image.

The algorithm is used to determine the homography to obtain tag attitude,

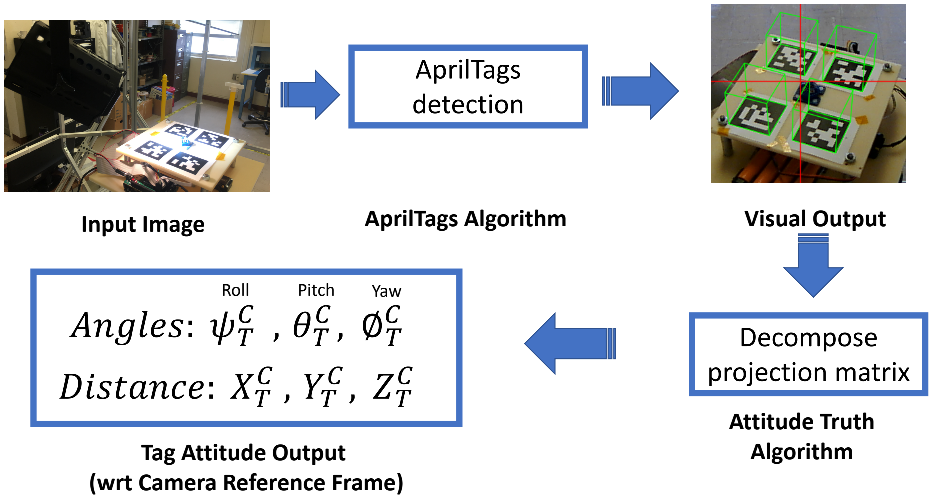

Input Image
AprilTags detection
AprilTags Algorithm
Visual Output
Decompose projection matrix
Attitude Truth Algorithm
Roll
Pitch
Yaw
$Angles: \psi_T^C\ , \theta_T^C, \emptyset_T^C$
$Distance: X_T^C\ , Y_T^C\ , Z_T^C$
Tag Attitude Output
(wrt Camera Reference Frame)

Figure 2.4: Overview of the mechanics of the OpenCV AprilTag software using the AprilTag extrinsic visual markers for CubeSat tracking. The AprilTag software inputs camera characteristics and outputs the angles and cartesian coordinates of the tags in the camera reference frame

position, and distances in camera reference frame (See Fig. 2.4 to see the camera based attitude truth system architecture). Specifics of the localization method and the Direct Linear Transform algorithm used for AprilTag software detection and estimation are described in [17] and [18]. Computation of a tag's position and orientation requires information about the operating environment. Namely, the intrinsic camera matrix (focal length of the camera and frame size), distortion camera matrix (tangential and radial lens distortions), and the physical size of the tag are all essential for accurately locating and extrapolating attitude data of the tag. From a 2D image the physical location and rotational information for a specific tag are determined a priori using a combination of the AprilTag software package and the OpenCV open source computer vision software package. An additional appealing feature of the AprilTag algorithm is that multiple tags can be used together to improve both robustness and accuracy. Use of multiple tags helps to avoid loss of attitude data due to occlusions and are compatible with standard deviation filters or other attitude filtering strategies (i.e. extended Kalman filter).

2.3 Extrinsic Camera Calibration

The objective of the attitude truth system extrinsic calibration is to find the rotation transformation from the camera reference frame to the inertial reference frame using an external reference plate. The external reference is a laser-cut medium density fiber particle board (MDF) platform with an array of four 200 mm AprilTag markers (See Fig. 2.5). The external reference calibration protocol for this attitude truth system is developed to minimize offsets that occur due to repositioning and drift from the facility apparatus (e.g.Helmholtz cage system).

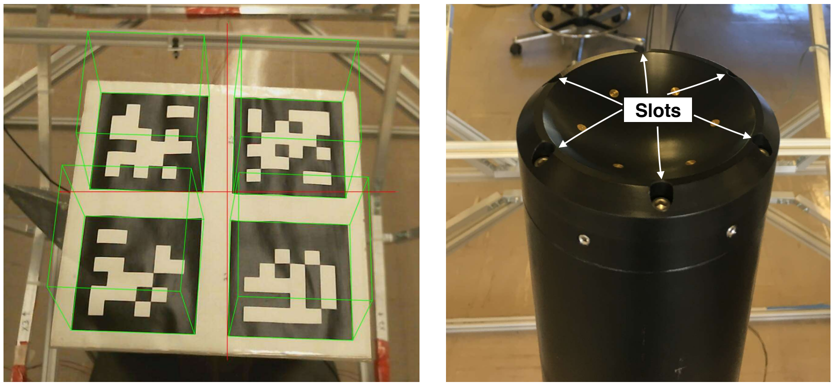
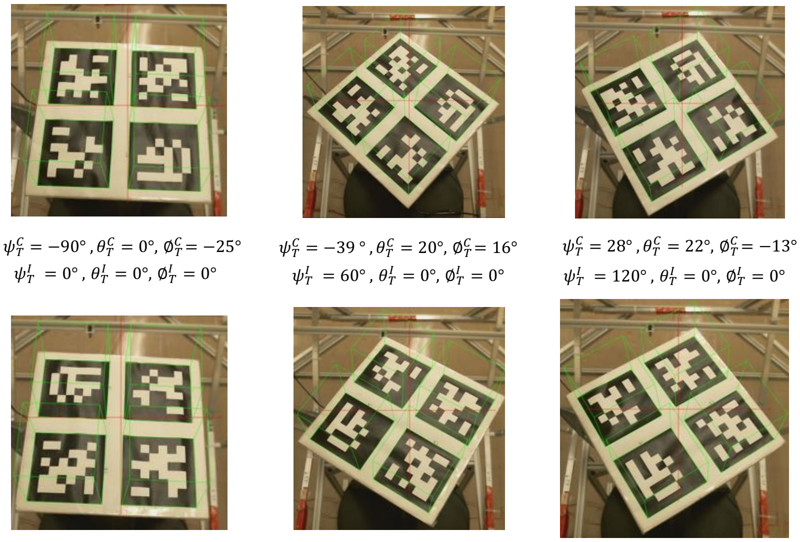


Figure 2.5: (Left) AprilTag calibration plate conjugate from the point of view of the camera with green cube outlines identifying the estimated AprilTag marker pose in real-time. (Right) The air bearing tower without the calibration plate from the point of view of the camera.

The AprilTag calibration plate has 6 pins on the bottom radially spaced 60°

apart to fit on top of the air-bearing tower slots. The 6 discrete positions provide 6 known angles of the calibration plate reference frame in the inertial reference frame.



$\psi_T^C = -90°, \theta_T^C = 0°, \emptyset_T^C = -25°$
$\psi_T^I = 0°, \theta_T^I = 0°, \emptyset_T^I = 0°$

$\psi_T^C = -39°, \theta_T^C = 20°, \emptyset_T^C = 16°$
$\psi_T^I = 60°, \theta_T^I = 0°, \emptyset_T^I = 0°$

$\psi_T^C = 28°, \theta_T^C = 22°, \emptyset_T^C = -13°$
$\psi_T^I = 120°, \theta_T^I = 0°, \emptyset_T^I = 0°$

$\psi_T^C = 90°, \theta_T^C = 0°, \emptyset_T^C = -26°$
$\psi_T^I = 180°, \theta_T^I = 0°, \emptyset_T^I = 0°$

$\psi_T^C = 150°, \theta_T^C = -23°, \emptyset_T^C = -13°$
$\psi_T^I = -120°, \theta_T^I = 0°, \emptyset_T^I = 0°$

$\psi_T^C = -150°, \theta_T^C = -23°, \emptyset_T^C = 13°$
$\psi_T^I = -60°, \theta_T^I = 0°, \emptyset_T^I = 0°$

Figure 2.6: The AprilTag calibration plate is rotated into 6 different positions about the z-axis with the four 200 mm visual marker tags in the field of view of the camera. Each snapshot of the calibration plate has information about the tags with respect to the camera reference frame (i.e. ψ_T^C, θ_T^C,ϕ_T^C) and the inertial frame (i.e. ψ_T^I, θ_T^I,ϕ_T^I).

The calibration process takes estimated angles of the calibration plate with respect to the camera reference frame (i.e. ψ_T^C, θ_T^C,ϕ_T^C) and known angles of the calibration plate in the inertial frame (i.e. ψ_T^I, θ_T^I,ϕ_T^I) to determine the rotation of the camera reference frame in the inertial reference frame ($[R]_C^I$). The process takes the roll, pitch and yaw measurements of the AprilTag markers in the camera reference frame, represented as ψ_T^C, θ_T^C and ϕ_T^C and determines the characteristic rotation matrix of fiducial tags in the camera reference frame ($[R]_T^C$) using equations 3.14 from reference [9].

After obtaining $[R]_T^C$ of the visual marker tags on the ADCS CubeSat testbed in the camera reference frame, the rotation matrix of the tags in the inertial frame ($[R]^I_T$) needs to be determined using the known angles in roll, pitch and yaw of the calibration plate in the inertial reference frame. Similarly, the rotation matrix is calculated for each known rotation mapping of $\psi^I_T = 0, 60, 120, 180, 240, 300$ (seen in Fig. 2.6). The full procedural calibration scheme as reflected with the rotation snapshots starting with the known rotation position of the tags $\psi^I_T = 0°$, $\theta^I_T = 0°$ and $\psi^I_T = 0°$ are detailed below:

> 1) Place calibration plate on top of frictionless air bearing seated into the slots. (see Fig. 2.6)
>
> 2) User will be prompted to input the known ϕ^I_T, θ^I_T and ψ^I_T for that round of the calibration. The GNAT AprilTag attitude truth system calibration C++ program then determines the best estimated $[R]^C_T$.
>
> 3) Repeat steps 1-2 for all 6 calibration plate placements [0°,60°,120°,180°,240°, 300°] as seen in Fig. 2.6 to obtain $[R]^I_C(i)$. Where i is the index of the angle.
>
> 4) The program with the determined $[R]^C_T(i)$ and $[R]^I_T(i)$ for all six iterations estimates the camera pose in the inertial reference frame based on methods explained in reference [16]. The estimated $[\hat{R}^I_C]$ is saved as the calibration value for a single camera of the G-NAT attitude truth system.

The extrinsic calibration allows users to position the camera anywhere around the test apparatus facility and have accurate ϕ^I_T, θ^I_T and ψ^I_T attitude results and to avoid occlusions (i.e. from the sun emulator).

2.4 Attitude Truth System with Extended Kalman Filter (ATS EKF)

The attitude truth system implementation overview seen in Fig. 2.7 maps the rotations from the camera reference frame to the inertial reference frame and uses an

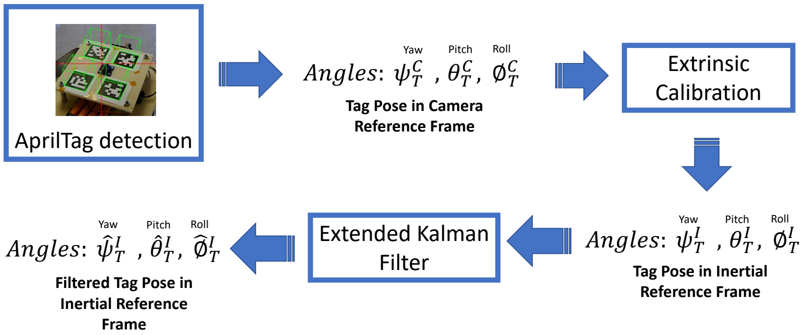

AprilTag detection
Yaw Pitch Roll
$Angles\colon \psi_T^C, \theta_T^C, \emptyset_T^C$
Tag Pose in Camera Reference Frame
Extrinsic Calibration
Yaw Pitch Roll
$Angles\colon \psi_T^I, \theta_T^I, \emptyset_T^I$
Tag Pose in Inertial Reference Frame
Extended Kalman Filter
Yaw Pitch Roll
$Angles\colon \hat{\psi}_T^I, \hat{\theta}_T^I, \hat{\emptyset}_T^I$
Filtered Tag Pose in Inertial Reference Frame

Figure 2.7: Attitude truth system leverages the AprilTag detection algorithm to obtain pose of tags on the testbed in the camera reference. Those tag angles ψ_T^C, θ_T^C and ϕ_T^C are then mapped from the camera reference frame to the inertial reference frame using the parameters determined by the extrinsic calibration. Finally the calibrated tag angles are passed through an extended Kalman filter to identify outliers and output accurate estimate attitude of the testbed.

extended Kalman filter (EKF) to filter out outliers and interpolate between high variance data. According to the AprilTag library [17], [18] the attitude accuracy from the current architecture is $\pm \leq 0.5^\circ$ in roll, pitch and yaw when tags are stationary. During our facility tests when the testbed is moving the AprilTag data standard deviation increases and accuracy decreases significantly. The AprilTag tag pose estimation can be effected by occlusions, light variability and especially motion. When the testbed is moving the attitude tag data error can reach levels of $\pm 20^\circ$ in ψ, θ and ϕ. The inaccuracy makes the fiducial marker system less than ideal for attitude tracking, but this can be improved with an EKF estimate of Euler angles and body rates. The EKF also identifies data outliers and interpolates when the data is sporadic and/or noisy.

The non-linear dynamical model and measurement model mentioned in Section 1.4 provides context to understand the non-linear discrete time model used in this implementation. The ξ_p, ξ_q, and ξ_r zero mean Gaussian noise has the spectral density matrix Q. The white random sequence of zero mean Gaussian random variables, v_θ, v_ϕ, and v_ψ has an associated covariance matrix R_k [13].

The process model noise variance matrix Q was determined through trial and error to be:

$$Q = \begin{bmatrix} 2.5 & 0 & 0 \\ 0 & 2.5 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.1}$$

The measurement data noise when the tags are moving can reach values of $\pm 20^o$ in roll, pitch and yaw. The noise covariance matrix R for use in the measurement model is described by the matrix $R = E\{\underline{v}\underline{v}^T\}$. When structured R has $(v_{\psi_m})^2$, $(v_{\theta_m})^2$ and $(v_{\phi_m})^2$ on the diagonals describing the roll, pitch and yaw observation covariances, respectively.

$$R_k = \begin{bmatrix} (v_{\phi_m})^2 & 0 & 0 \\ 0 & (v_{\theta_m})^2 & 0 \\ 0 & 0 & (v_{\psi_m})^2 \end{bmatrix} \tag{2.2}$$

The EKF has the ability to interpolate between the known measurements despite the possibility of measurements being sparse and/or highly variant. The full implementation of the attitude truth system EKF process is as follows:

Beginning with an initial estimate of the state and covariance, at each time step the state is propagated using the prediction update:

1) Propagate the state ($\hat{\underline{x}}^-_{k+1}$)

$$\hat{\underline{x}}^-_{k+1} = \Phi \hat{\underline{x}}^+_k \tag{2.3}$$

2) Propagate the error covariance ahead ($\hat{\underline{P}}^-_{k+1}$)

$$\hat{\underline{P}}^-_{k+1} = \Phi \hat{\underline{P}}^+_k \Phi^T + \Gamma Q \Gamma^T \tag{2.4}$$

The error covariance, state predictions and camera data are passed through a measurement update sequence:

1) Compute the Kalman Gain($K(k)$)

$$K_k = \underline{\hat{P}}^-_{k+1} H^T (H \underline{\hat{P}}^-_{k+1} H^T + R)^{-1} \tag{2.5}$$

2) Calculate the residual $r_k = (y_k - H(\underline{\hat{x}}^-_{k+1})$

3) Determine data outliers (leveraging known constraints and Kalman error thresholds) using the Outliers Determination using Kalman Residual Check algorithm (Algorithm 7). The EKF has the ability to interpolate between the known measurements despite the possibility of measurements being sparse and/or noisy. Additionally, the roll and pitch measurements should never be more than ±15°, due to the constraints from the air bearing tower.

Algorithm 1: Outliers Determination using Kalman Residual Check

```
$K_{err} = diag\sqrt{H \underline{\hat{P}}^+_k H^T + R}$
if $r(t,:) \geq 2K_{err}(:)$ $OR$ $|y(1,k)| \geq 15^\circ$ $OR$ $|y(2,k)| \geq 15^\circ$ $OR$
  $|y(3,k)| \geq 180^\circ$ then
    $H = [0]_{[3x6]}$
    Flag data
else
    $H = [I_{[3\times3]} 0_{[3\times3]}]$
end
```

4) After running Algoritm 7, update the state ($\underline{\hat{x}}^+_{k+1}$)

$$\underline{\hat{x}}_{k+1}^{+} = \underline{\hat{x}}_{k+1}^{-} + K_k r_k \tag{2.6}$$

5) Update the error covariance ($\underline{\hat{P}}_{k+1}^{+}$)

$$\underline{\hat{P}}_{k+1}^{+} = (I - K_k H)\underline{\hat{P}}_{k+1}^{-} \tag{2.7}$$

The initial results of using the EKF on the AprilTag fiducial attitude data for roll, pitch and yaw are shown in Fig. 2.8. The roll and pitch EKF standard deviation converges to ±1.30° and ±0.61° for yaw after a few time steps and while the recorded residuals show fluctuations the EKF omits any residuals that are larger than $2K_{err}$ for state variables x_1, x_2, and x_3. The EKF dynamically changes its update step to interpolate without the information gained from the resulting measurement based innovation step and relies entirely on the non-linear model (See Outliers Determination using Kalman Residual Check algorithm).

The initial results of using the EKF on the AprilTag fiducial attitude data for estimated roll rate, pitch rate and yaw rate are determined using equation 4.70 in reference [9] with the estimated Euler angles and body rates (see Fig. 2.9). The time series estimated roll and pitch angular rates coincide with the fairly minor oscillatory movements of the ADCS testbed during mainly yaw rotations. The yaw angular rate is fairly steady over the several yaw rotations traveling at speeds up to 20 deg/s on the frictionless air bearing tower.

The EKF shows its ability to suppress sporadic noise from the AprilTag camera data and obtain a more accurate estimation of AprilTag attitude data than previously obtained for moving objects. Additionally, the EKF is able to determine estimated body

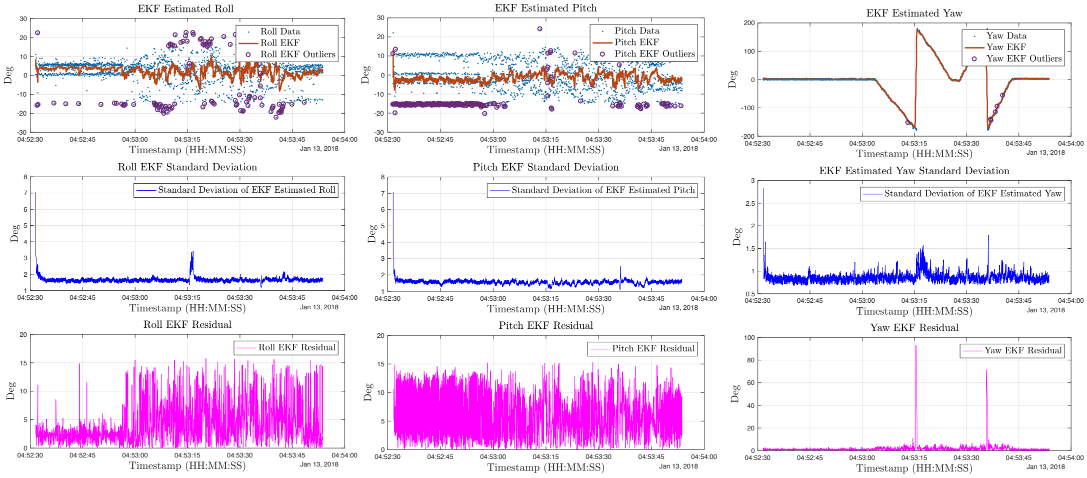


Figure 2.8: The top plots are of time series attitude truth system data (in black dots) and a smooth attitude estimation with the extended Kalman filter (in red). In purple are outliers of roll, pitch and yaw identified by the Outlier Determination using Kalman Residual Check algorithm. The middle plots are corresponding standard deviations of each estimated Euler angle of the ADCS testbed. The bottom plots are the time series of Kalman residuals for each estimated Euler angle of the ADCS testbed.

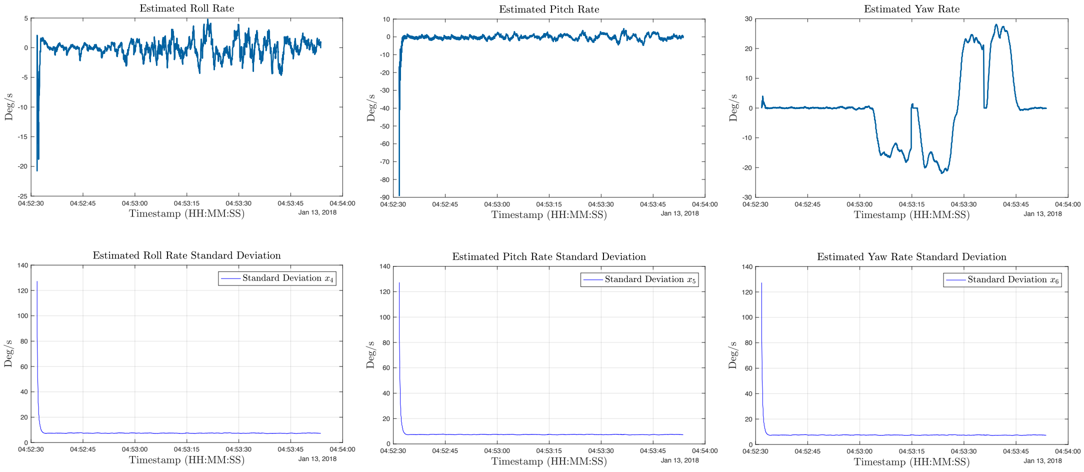


Figure 2.9: Time series attitude truth system data (in black dots). In the top plot in the blue is a smooth ATS attitude estimation of angular rates with the extended Kalman filter. The bottom plots are the corresponding standard deviations of each estimated roll rate, pitch rate and yaw rate of the ADCS testbed.

rates with only estimated AprilTag camera attitude data. The performance of the EKF can be further evaluated in Table 2.2 by examining the standard deviation for each of the estimated states.

Table 2.2: Time Series Converged Standard Deviation of the EKF for all 6 State Variables

	SD at k=1	SD at k=3	SD at k=5	SD at k=1500	SD at k=last
ϕ	20	4.08	2.54	1.72	1.30
θ	20	4.08	2.53	1.28	1.30
ψ	20	1.52	1.03	0.89	0.61
p	115	28	9.39	4.95	4.49
q	114	28	9.43	4.93	4.38
r	114	13	4.86	3.27	3.47

The time series standard deviation provides us with enough insight to show that the noisy AprilTag based attitude truth system data can be improved by the use of the EKF (with the built-in Outliers Determination using Kalman Residual Check algorithm). The standard deviation of the data converges after the first few iterations and decreases to ±1.30°, ±1.30° and ±0.61° from ±20°, ±20° and ±20°, for each Euler angle.

The attitude truth system accuracy and robustness improvements were proven to rectify noisy attitude observations during quick maneuvers using an extended Kalman Filter. A general overview of the EKF running simultaneously on all three axis of rotation of the ADCS testbed can be found in Fig. 2.10. The AprilTag based attitude truth system data implemented with the extended Kalman filter was able to maintain estimation standard deviation levels lower than the standard deviation of raw AprilTag attitude data when tracking moving tags. The objective of this lab is to advance the technology readiness level of ADCS and avionics technologies for CubeSat missions under development by NASA and its partners in academia and industry. The functioning testbed environment using the extended Kalman filter AprilTag based attitude truth system in the G-NAT laboratory meets CubeSat mission specifications for testing and verifying ADCS testbeds.

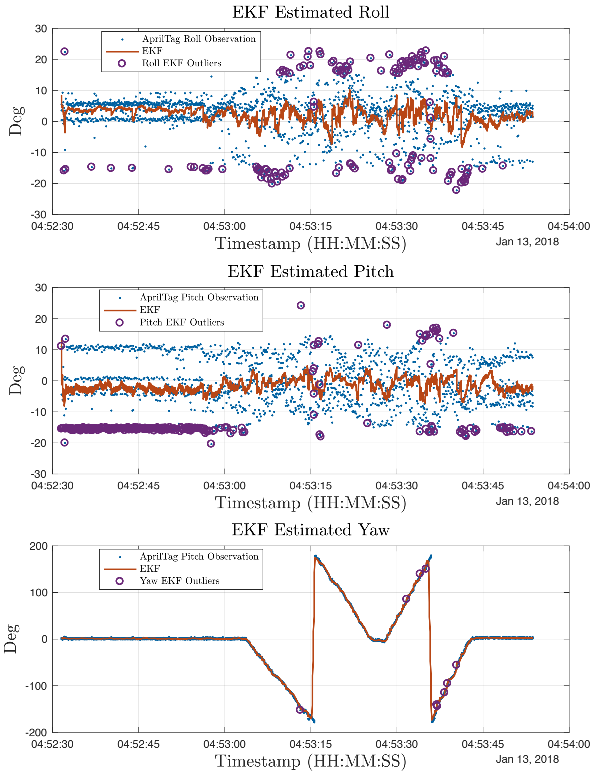


Figure 2.10: Time series ATS data (in black dots) where the EKF filters is investigated. In red is a smooth ATS signal with the extended Kalman filter. In purple are outliers identified by the Outlier Determination using Kalman Residual Check algorithm. The top plot is the estimated roll of the ADCS testbed oscillating around zero. The middle plot is the estimated pitch of the ADCS testbed oscillating around zero. The bottom plot is the yaw of the ADCS testbed rotating counter-clockwise about body z-axis

Chapter 3

CubeSat Testbed System

3.1 Introduction

3.1.1 Introduction to Attitude Determination

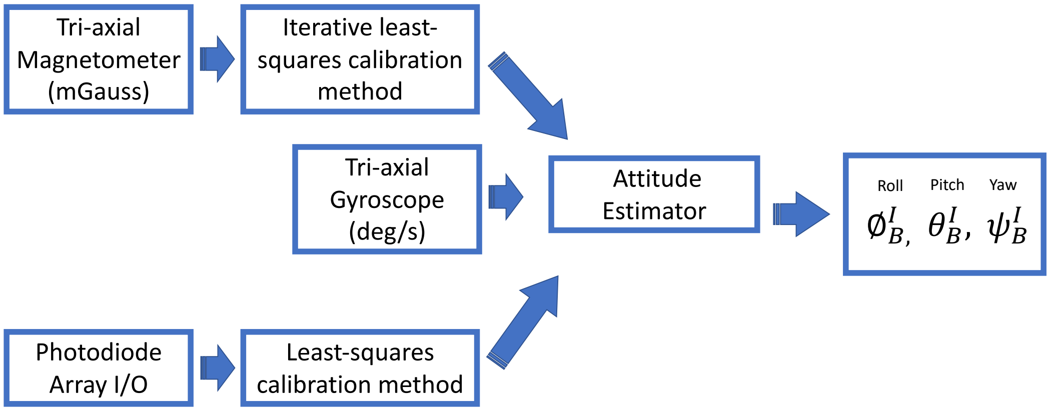

Tri-axial Magnetometer (mGauss)
Iterative least-squares calibration method
Tri-axial Gyroscope (deg/s)
Attitude Estimator
Photodiode Array I/O
Least-squares calibration method
Roll
Pitch
Yaw
$\phi_B^I, \theta_B^I, \psi_B^I$

Figure 3.1: ADCS Attitude Estimation Topology: The output of the attitude estimation algorithms for this study are Euler angles of the testbed body reference frame in the inertial reference frame.

The purpose of on-board attitude determination is to track an objects motion and orientation in space with respect to a global (i.e. inertial) reference frame using on-board sensors. For this research study a magnetometer (used to measure the geomagnetic field vector), gyroscope (used to measure body angular rate) and photodiodes (used to measure sun direction vector) are reference inputs for the two attitude estimation algorithms of this study. The on-board testbed sensor system topology used for the CubeSat ADCS system is outlined in Fig. 3.1 showing the general the attitude determination process.

Section 3.2.1 details the process of calibrating the photodiode tetrahedron into a normalized tri-axial sensor using a weighted least-squares approximation. The result of the calibration is a tri-axial reference vector to the sun (Or in our case a sun emulator bulb).

Section 3.2.2 -3.2.4 will explain the two-step iterative least-squares method to calibrate the testbed ADCS magnetometer included in the MEMS inertial measurement unit (IMU) seen on the bottom deck of the testbed. The calibration of the magnetometer output must be performed to account for scaling, centering and non-linearities of the sensors.

Section 3.3 will explain the batch method for solving the misalignment of multiple heterogeneous sensor reference frames (e.g. magnetometer reference frame and sun sensor reference frame) using tumble data [19]. The batch misalignment will determine the rotation transformation needed to align the sun sensor reference frame to the magnetometer reference frame.

Finally, section 3.4 will explain the implementation of two attitude estimation algorithms on the ADCS and verified by the ATS with EKF. The first attitude estimator, Tri-axial attitude determination (TRIAD) takes solely the tri-axial sun sensor vector and magnetometer vector to produce a DCM representing rotation of the ADCS body reference in the inertial reference frame. The second attitude estimator, the explicit complementary filter uses the tri-axial sun sensor vector and magnetometer vector to correct the rotation estimation propagation using angular rates from the on-board gyroscope.

3.2 Sensor Calibration

3.2.1 Sun Sensor Calibration

In the GNAT lab we utilize an array of photodiodes to determine a tri-axial reference vector to a sun emulator bulb as an additional input to our attitude estimation process. As seen in the Fig. 3.2 a tetrahedron of sun sensors is mounted on the top deck of the testbed for measuring the sun vector.

Ideally, a photodiode produces current I as a function of incoming light according to the model [1]

$$I = I_0 cos(\theta) \tag{3.1}$$

where θ is the angle between the direction normal to the photosensitive plane and the line-of-sight vector to the sun (herein referred to as sun vector), and I_0 is the maximum current output of the sensor, corresponding to $\theta = 0$. A weighted least square

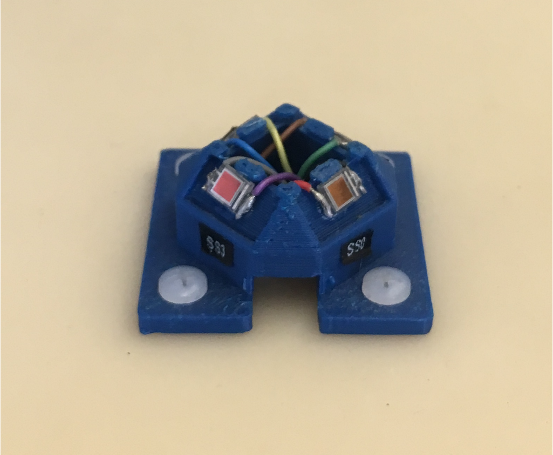


Figure 3.2: Photodiode tetrahedron placed on top of the testbed ADCS system.

approximation is used to minimize the angular uncertainty of the sun vector estimate as a function of the photodiode orientation and variance of the sensor as seen in Fig. 3.3.

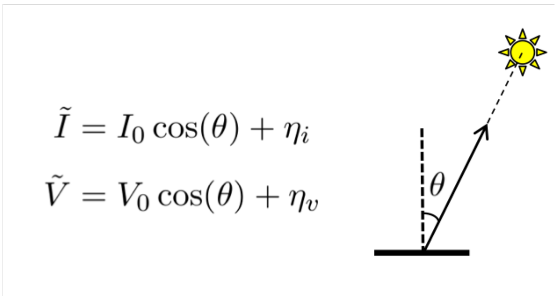

$\tilde{I} = I_0 \cos(\theta) + \eta_i$
$\tilde{V} = V_0 \cos(\theta) + \eta_v$
θ

Figure 3.3: Sun photodiode array model as function of cosine of the direct irradiance angle [1]

The sun vector estimation is formulated as finding the intersection of multiple planes, where the planes are defined by the photodiode normal directions and the measured currents. The formulation is illustrated for a two-dimensional case in Fig. 3.3. In Fig. 3.3, the rectangles represent two photodiodes, the dashed arrows show the directions normal to the photosensitive plane, $\hat{n}_i$ (where $i \in 1, 2$ is the photodiode index), the solid arrow shows the sun vector, $\hat{s}$, and θ_i is the angle between the photodiode normal direction and the sun vector, as in equation 3.1. The sun vector $\hat{s}$ corresponds to the intersection point of the two planes, shown by the dotted lines, and $\hat{s}$ is estimated by finding the intersection point of the two planes. The planes are perpendicular to the photodiode normal direction, and, referring to, their location along the normal direction $\hat{n}_i = \frac{I_i}{I_{0,i}}$. In general, a plane can be defined by a known point on the plane, r_0, and a normal vector to the plane, p, according to equation 3.2, where r is the location of any point on the plane.

$$p^T(r - r_0) = 0 \tag{3.2}$$

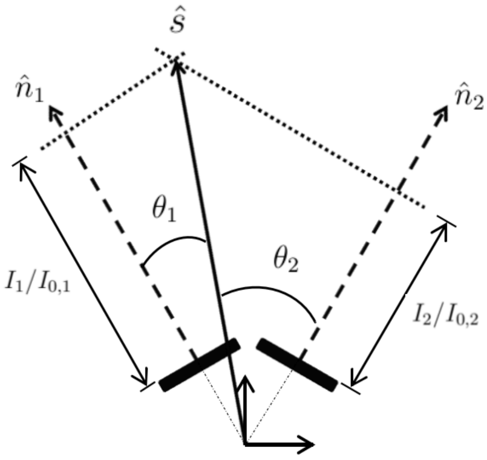


Figure 3.4: The solid rectangles represent the two photodiodes, and their normal directions are shown by the unit vectors n_i, $i \in 1, 2$. The angles θ_i define the sun vector direction relative to the photodiode normal directions, as in equation 3.1 [2]

In application the photodiode configuration in Fig. 3.4 becomes

$$\underline{n}_i^T(\underline{s} - \frac{I_i}{I_{0,i}}\underline{n}_i) = 0 \tag{3.3}$$

where $\underline{n}_i = [n_{i,1}, n_{i,2}, n_{i,3}]^T$ and $\underline{s} = [s_1, s_2, s_3]$ as long as $||\underline{n}|| = 1$. The sun vector $\underline{s}$ corresponds to the intersection of the two planes or the weighted contributions from each photodiode direction. The planes are represented by the dotted lines which are perpendicular to $\underline{n}_i$, and their location along $\underline{n}_i$, which is given by $\frac{I_i}{I_{0,i}}$. In summary,

$$\frac{I_i}{I_0} = \begin{bmatrix} n_{i,1} & n_{i,2} & n_{i,3} \end{bmatrix} \begin{bmatrix} s_1 \\ s_2 \\ s_3 \end{bmatrix} \tag{3.4}$$

Given 4 non-parallel and non-coplanar photodiodes arranged in the tetrahedron on the top platform of the testbed, the sun vector estimate equation is $\underline{y} = [H]\underline{s} + \underline{\mu}$, where $\underline{s}$ is to be determined by a least squares approximation. μ is zero mean Gaussian white noise of the individual photodiodes. The known components in matrix form are

$$\underline{y} = \begin{bmatrix} \frac{\tilde{I}_1}{I_{0,i}} \\ \\ \frac{\tilde{I}_2}{I_{1,i}} \\ \\ \frac{\tilde{I}_3}{I_{3,i}} \\ \\ \frac{\tilde{I}_4}{I_{4,i}} \end{bmatrix} \quad [H] = \begin{bmatrix} n_{1,1} & n_{1,2} & n_{1,3} \\ n_{2,1} & n_{2,2} & n_{2,3} \\ n_{3,1} & n_{3,2} & n_{3,3} \\ n_{4,1} & n_{4,2} & n_{4,3} \end{bmatrix} \quad \underline{\mu} = \begin{bmatrix} \frac{\mu_1}{I_{0,1}} \\ \\ \frac{\mu_2}{I_{0,2}} \\ \\ \frac{\mu_4}{I_{0,4}} \end{bmatrix} \tag{3.5}$$

The exact H for the tetrahedon photodiode structure based on the unit vector contributions are seen in equation 3.6, because the photodiodes are each positive 45 degrees of the testbed plane and each are 90° rotations about the z-axis of each other. In application H is,

$$H = \begin{bmatrix} \sqrt{3}/3 & \sqrt{3}/3 & -\sqrt{3}/3 \\ -\sqrt{3}/3 & \sqrt{3}/3 & -\sqrt{3}/3 \\ -\sqrt{3}/3 & -\sqrt{3}/3 & -\sqrt{3}/3 \\ \sqrt{3}/3 & -\sqrt{3}/3 & -\sqrt{3}/3 \end{bmatrix} \tag{3.6}$$

The uncertainty of the sun vector estimate is a function of the variance of the individual photodiodes as well as the photodiode configuration. The expected value of the variances creates a covariance matrix R where $R = E\{\mu^T \mu\}^2$. Using a weighted least-squares approximation $\hat{s}$ can be solved granted A is full rank as illustrated in equation 3.7.

$$\underline{\hat{s}} = (H^T R^{-1} H)^{-1} H^T R^{-1} \underline{y} \tag{3.7}$$

The approximated $\hat{s_1}$, $\hat{s_2}$, $\hat{s_3}$ become the normalized tri-axial sun vector input as the attitude estimator.

The sun sensors resultant unit vector components for x, y, and z can be seen in Fig. 3.5.

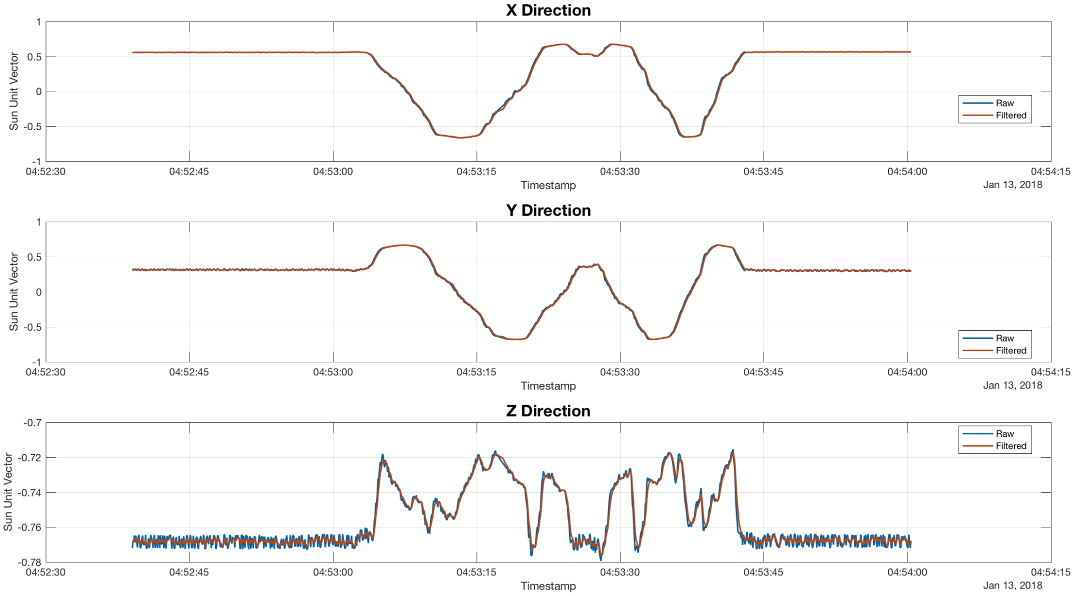


Figure 3.5: The determined sun tri-axial x, y, and z reference vector components with the sun pointed 52.6° off of the xy inertial reference plane.

For this CubeSat testbed case study the testbed unit was rotated clockwise 360 degrees and then immediately counter clock-wise 360 degrees and the sun emulator bulb was tilted pointing towards the center of the testbed at a 37.4° degrees. The angle of determined sun sensor vector as seen in Fig. 3.5 was able to estimate that sun direction off the x-y plane.

3.2.2 IMU Calibration

The MEMS inertial measurement unit (IMU) used for this study, seen in Fig. 3.6 has a suite of sensors including an accelerometer, gyroscope and magnetometer. The attitude determination research performed only uses two of the three sensors, the magnetometer and gyroscope. These sensors in this IMU are not initially calibrated and thus a calibration must be performed. Calibration of the magnetometer must be performed to account for scaling, centering ,and non-linearities for the magnetometer. Calibration of the gyroscope must be determined to account for scaling and bias null shift.

Figure 3.6: Analog MEMS IMU sensor

Traditionally, in inertial navigation there exists various methods for three-axis sensor (i.e. gyroscopes, magnetometers) calibration that require expansive tools. The setup can be demanding to acquire the data ,and compare them against a fixed reference. Usually IMU calibration is achieved using a calibration table whose varying orientation is precisely measured. [20] These processes can be expensive. Most CubeSat projects using low-cost MEMS IMU sensors do not have the access to these resources. A simple, but effective iterative calibration method is described in reference [20] is performed on the IMU three-axis magnetometer on-board the CubeSat testbed.

3.2.3 Magnetometer Calibration and Setup

Method for obtaining a full data set for calibration requires taking the ADCS testbed with the current on-board sources of magnetic field noise (from hard and soft irons sources) present on the testbed and rotating the device in all 3 DOF to obtain tumble data. Raw tri-axial magnetometer measurements are denoted as m_i (3 by 1 vector), where i stands for the index and the ideal/actual magnetometer vector is denoted as $\hat{m_i}$ (3 by 1 vector). The raw measurements values m_i are scaled with matrix A and centered with zero-bias vector B.

$$\underline{\hat{m}} = A\underline{m} + \underline{B} \tag{3.8}$$

where A is a scaling matrix, $\underline{\hat{m}}$ is the ideal/actual magnetic field unit vector, and $\underline{B}$ is the measurement null shift vector.

Obtaining A and B requires fitting a 3-D ellipsoid of $\underline{m}$ into a 3-D unit ball where $||\underline{\hat{m}}||^2 = 1$ using an iterative least squares method. Subject to the cost function $||A(\underline{\hat{m}} - B)||^2 = 1$ for every sample i=1,...,n. Where n is the number of measurements. Expanding the equation a cost function $g(A, B)$ can be created to minimize g and determine A, B. [20].

$$g(A,B) = \Sigma(||A(\underline{m} - B)||^2 - 1)^2 \tag{3.9}$$

Using the non-linear, two-step estimation algorithm presented in reference [20] one can obtain the calibration scaling A or zero-bias offset B vector for the magnetometer using the modified cost function:

$$h(A,B,k) = \sum_{i=1}^{n}(||A(\underline{m}_i(k) - B)||^2 - \frac{\hat{\underline{m}}_i(k)}{||\hat{\underline{m}}_i(k)||})^2 \tag{3.10}$$

Minimize h by iteratively solving the least-squares method on all the sampled data over k iterations. The above equation can be reconstructed to fit within the least-squares approximation setup $y = Tx$. Equation 3.11-3.12 is the reconstruction of equation 3.10 into the least squares model.

$$T = \begin{bmatrix} m_{1,i}(k) & 0 & 0 & \dots & m_{1,n}(k) & 0 & 0 \\ m_{2,i}(k) & 0 & 0 & \dots & m_{2,n}(k) & 0 & 0 \\ m_{3,i}(k) & 0 & 0 & \dots & m_{3,n}(k) & 0 & 0 \\ 0 & m_{1,i}(k) & 0 & \dots & 0 & m_{1,n}(k) & 0 \\ 0 & m_{2,i}(k) & 0 & \dots & 0 & m_{2,n}(k) & 0 \\ 0 & m_{3,i}(k) & 0 & \dots & 0 & m_{3,n}(k) & 0 \\ 0 & 0 & m_{1,i}(k) & \dots & 0 & 0 & m_{1,n}(k) \\ 0 & 0 & m_{2,i}(k) & \dots & 0 & 0 & m_{2,n}(k) \\ 0 & 0 & m_{3,i}(k) & \dots & 0 & 0 & m_{3,n}(k) \\ 1 & 0 & 0 & \dots & 1 & 0 & 0 \\ 0 & 1 & 0 & \dots & 0 & 1 & 0 \\ 0 & 0 & 1 & \dots & 0 & 0 & 1 \end{bmatrix}^T \tag{3.11}$$

$$x = \begin{bmatrix} \tilde{A_{11}} \\ \tilde{A_{12}} \\ \tilde{A_{13}} \\ \tilde{A_{21}} \\ \tilde{A_{22}} \\ \tilde{A_{23}} \\ \tilde{A_{31}} \\ \tilde{A_{32}} \\ \tilde{A_{33}} \\ \tilde{B_1} \\ \tilde{B_2} \\ \tilde{B_3} \end{bmatrix}, y = \begin{bmatrix} \frac{m_{1,i}(k)}{||\underline{m}||} \\ \frac{m_{2,i}(k)}{||\underline{m}||} \\ \frac{m_{3,i}(k)}{||\underline{m}||} \\ . \\ . \\ . \\ \frac{m_{1,n}(k)}{||\underline{m}||} \\ \frac{m_{2,n}(k)}{||\underline{m}||} \\ \frac{m_{3,n}(k)}{||\underline{m}||} \end{bmatrix} \tag{3.12}$$

Now with T, x, y setup the estimated $\underline{\hat{x}}$ being can be formulated $\hat{x} = (T^T T)^{-1} T^T y$, beginning with an initial guess when $k = 1$; $\tilde{A}_o$ and $\tilde{B}_o$ below is the 2nd (or iterative) step where you calculate the next estimation of $\tilde{A}$ and $\tilde{B}$ which is a reshape of $\hat{x}$:

$$\tilde{A}^+ = \begin{bmatrix} \hat{x}_1 & \hat{x}_2 & \hat{x}_3 \\ \hat{x}_4 & \hat{x}_5 & \hat{x}_6 \\ \hat{x}_7 & \hat{x}_8 & \hat{x}_9 \end{bmatrix}, \tilde{B}^+ = \begin{bmatrix} \hat{x}_{10} \\ \hat{x}_{11} \\ \hat{x}_{12} \end{bmatrix} \tag{3.13}$$

The determined $\tilde{A}^+$ and $\tilde{B}^+$ become the input for the next step to obtain the next iterative calibrated version of the data $\underline{m}^+$

$$\underline{m}_{i:n}(k+1) = \tilde{A}^+ \underline{m}_{i:n}(k) + \tilde{B}^+ \tag{3.14}$$

$$\tilde{B}(k+1) = \tilde{A}^+ \tilde{B}(k) + \tilde{B}^+ \tag{3.15}$$

$$\tilde{A}(k+1) = \tilde{A}^+ \tilde{A}(k) \tag{3.16}$$

After a few k iterations (2-20) the matrix $\tilde{A}_k$ and $\tilde{B}_k$ are determined and become $\tilde{A}_{final}$ and $\tilde{B}_{final}$. The magnetometer measurements are then calibrated using the final $\tilde{A}_{final}$ and $\tilde{B}_{final}$.

$$\underline{\hat{m}} = \tilde{A}_{final} \underline{m} + \tilde{B}_{final}$$

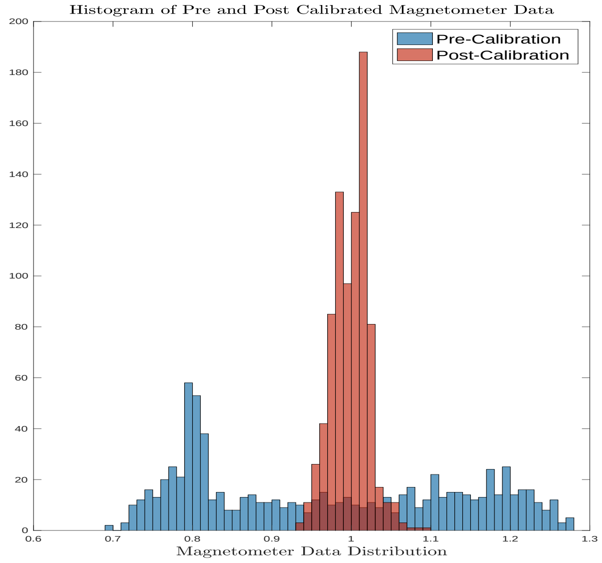


Figure 3.7: Histogram of the pre-calibrated and post-calibrated magnetometer normalized data. The post-calibrated magnetometer data improvement over the non-calibrated magnetometer data can be seen by the more narrow distribution.

The $\underline{\hat{m}}$ is the calibrated magnetometer tri-axial input that becomes the input for the attitude estimator.

The determined $\tilde{A}$ and $\tilde{B}$ from algorithm above on the testbed data are as follows:

$$\tilde{A} = \begin{bmatrix} 0.0025 & 0.0001 & -1.306e-05 \\ -0.0001 & 0.0026 & -6.716e-6 \\ 1.315e-05 & 5.680e-05 & 0.0026 \end{bmatrix}, \tilde{B} = \begin{bmatrix} -0.2162 \\ 0.0492 \\ -0.1341 \end{bmatrix} \tag{3.17}$$

The determined $\tilde{B}$ value sources can be contributed to the on-board magnetic field being created from the battery systems, micro-controller and wiring. The precision improvement of the values can be seen by a histogram of both values overlaid as seen in Fig. 3.7

In Fig. 3.8 it shows the normalized calibrated magnetometer and the nor-

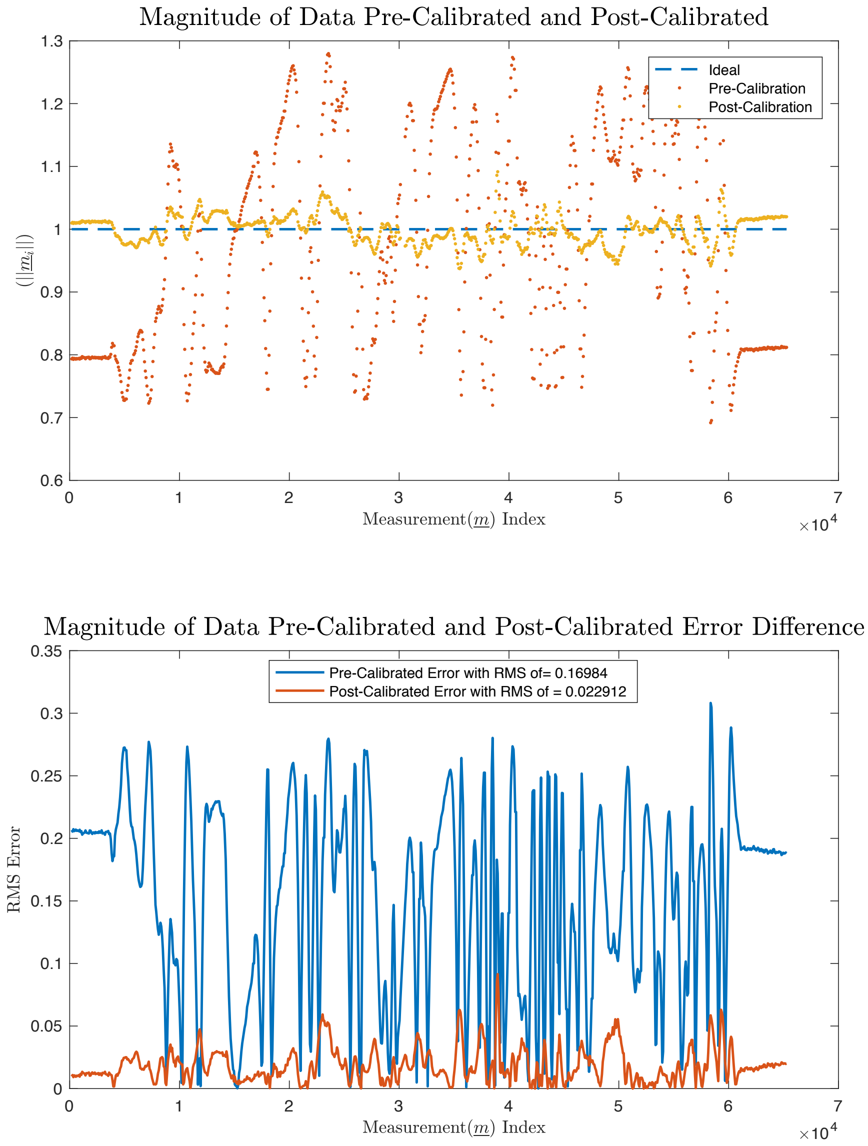


Figure 3.8: Normalized pre-calibrated and post-calibrated magnetometer data normalized. The RMS error between the post-calibrated magnetometer data and the ideal normalized value is 0.03 which is a roughly 6 times better than the RMS error between the pre-calibrated magnetometer data and the ideal normalized value.

malized uncalibrated magnetometer data. The bottom plot of Fig. 3.8 shows the RMS error $\epsilon_{calibrated} = |||\underline{m}_{calibrated,i}|| - 1|$ and the RMS error of the uncalibrated

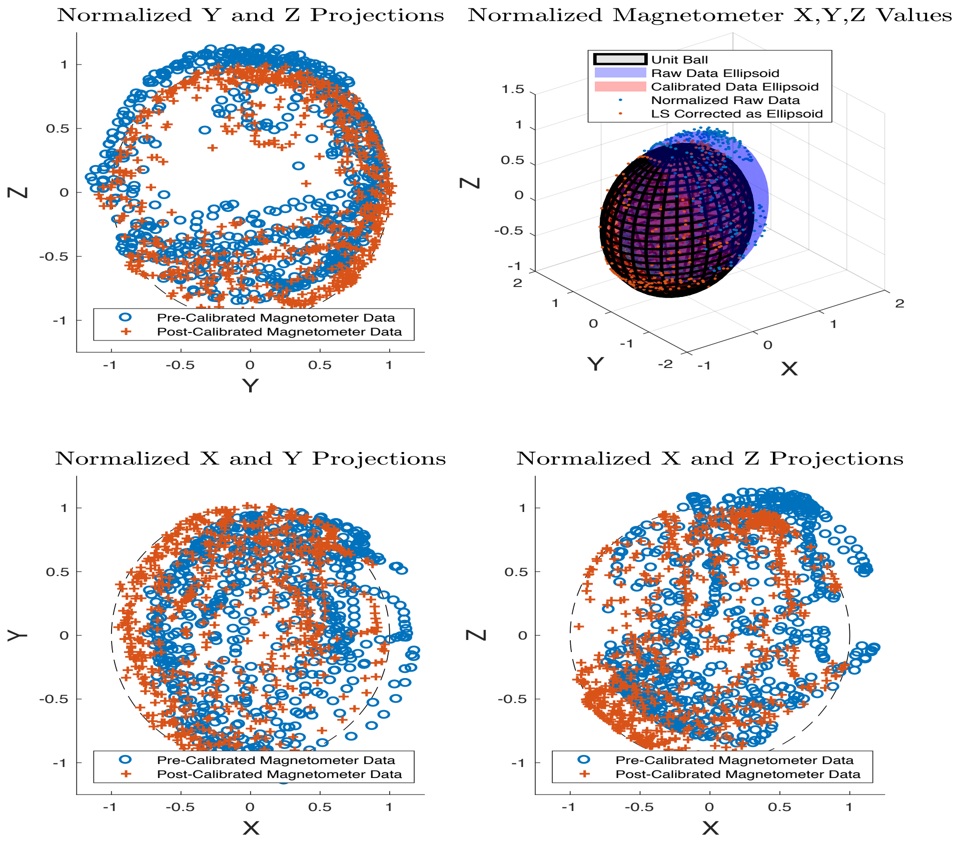


Figure 3.9: Normalized pre-calibrated and post-calibrated magnetometer data plotted on a 3D plot. The pre-calibrated magnetometer data in the blue sphere is offset significantly from the ideal unit ball

$\epsilon_{raw} = |||\underline{m}_{raw,i}|| - 1|$. The mean RMS error of the post-calibrated magnetometer data is almost 0.03 which is roughly 6 times better than the mean RMS error of the pre-calibrated magnetometer data.

The Fig. 3.9 shows the the unit ball with the calibrated values mapped on the surface of a unit ball mesh. The figure shows a significant influence of having a calibrated magnetometer (represented as red +'s) versus a non-calibrated magnetometer (represented as blue circles). The largest calibration contribution came from the recentering on the data most obvious in the x and z direction.

3.2.4 Gyroscope and Setup

The gyroscope being used for this study measures body angular velocity (rad/s) of the ADCS testbed and ideal body angular rate $\underline{\hat{\omega}}$ is interpreted as

$$\underline{\hat{\omega}} = a\underline{\omega} + \underline{b} \tag{3.18}$$

where $\underline{\omega}$ is the measured gyro values (rad/s), $\underline{\hat{\omega}}$ is the ideal gyro values, a is the scaling matrix, $\underline{b}$ is the bias vector. The bias vector $\underline{b}$ can change over time,

$$\underline{b} = \underline{b}_o + \underline{b}(t) \tag{3.19}$$

$\underline{b}_o$ = static null shift $\underline{b}(t)$ = gyro drift

The gyro drift $\underline{b}(t)$ error will slowly compound over integration of the angular rate values and the complementary filter is designed to suppress these gyro idiosyncrasies. The complementary filter can correct the drift from IMU MEMS gyroscopes while simultaneously determining the ADCS testbed Euler angle.

3.3 Sensor Alignment

3.3.1 Benefits of Aligning Sensor Reference Frames

Many individual sensors within integrated sensor systems inherently have non-coincident reference frames and this misalignment is very noticeable on our ADCS platform, specifically between the sun sensor reference frame and the magnetometer reference frame. Solving the misalignment problem for these heterogeneous sensors (e.g. magnetometer and sun sensor) can be performed with a batch approach using tumble data of the testbed and solving Wahba's problem [21]. In our case with the ADCS system we need to align the reference frame of the sun vector to the magnetometer reference frame, which has significant misalignments due to position and manufacturing errors.

The misalignment between the two reference frames can be described by understanding Wahba's problem where the assumption is made that there exists an optimal rotation between two normalized vectors

$$J(R) = \frac{1}{2}\sum_{i=1}^{n} a_i||w_i - Rv_i||^2 \tag{3.20}$$

where w_i are a set of vectors in one reference frame, v_i is the corresponding set of vectors in another reference frame, R is the rotation transformation from body to inertial frame. The a_i's are optional weights. The specific method to solve Wahba's Problem used in the batch sensor alignment algorithm is with the Markley solution [21].

$$B = \frac{1}{2}\sum_{i=1}^{n} a_i w_i v_i^T \tag{3.21}$$

$$svd(B) = U\Sigma V^T \tag{3.22}$$

$$R = UMV^T \tag{3.23}$$

where,

$$M = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & \det(U) & \det(V) \end{bmatrix} \tag{3.24}$$

The optimal rotation R from Markley's solution can be determined from any non-collinear vectors.

3.3.2 Misalignment Calculation

If the slave sensor(i.e. sun sensor) reference frame is not coincident with the master sensor (i.e. magnetometer) reference frame there exists a R_{mis} to rotate two sensors into the same reference frame as seen in reference [21]:

$$\underline{\hat{m}}_i^B = R_i^T[\underline{m}^I] + \nu_m \tag{3.25}$$
$$\underline{\hat{s}}_i^B = R_{mis}R_i^T[\underline{s}^I] + \nu_s \tag{3.26}$$

where ν(noise term), $\underline{m}^I$(master sensor in the inertial reference frame), $\underline{s}^I$(slave sensor in the inertial reference frame). Expanding equation 3.20 with equations 3.26, ignoring the noise terms and rearranging the equation the updated cost function is now:

$$J(R, R_{min}) = \frac{1}{2}\sum_{i=1}^{n} ||\underline{s}^I[I - R_i R_{mis} R_i^T]||^2 \tag{3.27}$$

Solving the above cost function with an iterative batch misalignment calibration method involves updating $\hat{R}_{mis}$ until the cost function value converges based on minimizing equation 3.27. The step-by-step batch misalignment algorithm from reference [21] is reiterated for this case study in Algorithm 2

After 20 iterations the Frobenious norm difference between $\hat{R}_{mis}^k$ and previous $\hat{R}_{mis}^{k-1}$ in Fig. 3.10 converges to roughly 0.001, which is significantly low enough to halt the algorithm and use the determined misalignment rotation between the two sensor reference frames.

The final misalignment rotation matrix from the sun sensor reference frame to the magnetometer reference frame

Algorithm 2: Compute R_{mis}

```
    Result: $\hat{R}_{mis}$
1   $\hat{R}_{mis} = I_{3\times3}$;
2   while $\hat{R}_{mis}$ not converged do
3       $[\hat{S}]^{B-} = \hat{R}^{-}_{mis}[\hat{s}^B_1, \hat{s}^B_2, \ldots, \hat{s}^B_i]$;
4       for $i \leftarrow 1$ to $n$ magnetometer/sunsensor measurements do
5           Solve Wahba's Problem using $v_i = [\underline{\hat{m}}^B_i, \underline{\hat{s}}^B_i]$ and $w_i = [\underline{m}^I, \underline{s}^I]$ for $R_{opttemp}(i)$ ;
6           $\hat{s}^B_i = R^T_{opttemp}\hat{s}^I$;
7           Collect all $\hat{s}^B_i$ into $[\hat{S}]^B = [\hat{s}^B_1, \hat{s}^B_2 \ldots \hat{s}^B_i]$;
8           Again solve Wahba's problem to solve the optimal rotation $\hat{R}^{+}_{mis}$ between $[\hat{S}]^{B-}$ and $[\hat{S}]^{B+}$ ;
9       end
10      if $||\hat{R}^{+}_{mis}$ - $\hat{R}^{previous+}_{mis}||_{Fronbenius} \leq$ tolerance then
11          Done;
12      else
13          go to step [3];
14      end
15  end
```

$$R_{mis} = \begin{bmatrix} -0.019756 & -0.96992 & -0.24262 \\ 0.9974 & -0.0023072 & -0.071994 \\ 0.069269 & -0.24341 & 0.96745 \end{bmatrix} \tag{3.28}$$

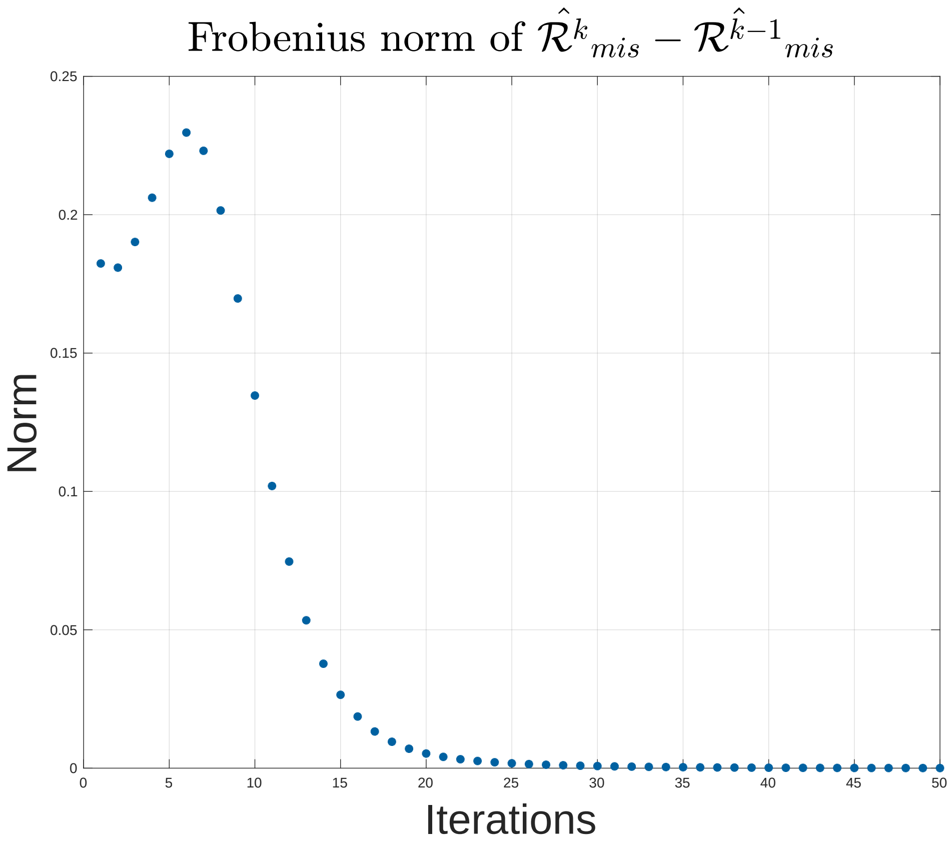


Figure 3.10: Convergence of the iterative batch alignment calibration algorithm. After 20 iterations the Frobenious norm difference between $\hat{R}^{k}_{mis}$ and previous $\hat{R}^{k-1}_{mis}$ converges to 0.001.

The significant changes of the determined batch alignment can be seen in Fig. 3.11 and there were small adjustments in the z axis component. Once the sun sensor's reference frame is coincident with the magnetometer reference frame the sun sensor inputs can be used as correct/aligned inputs for attitude estimation on the ADCS system.

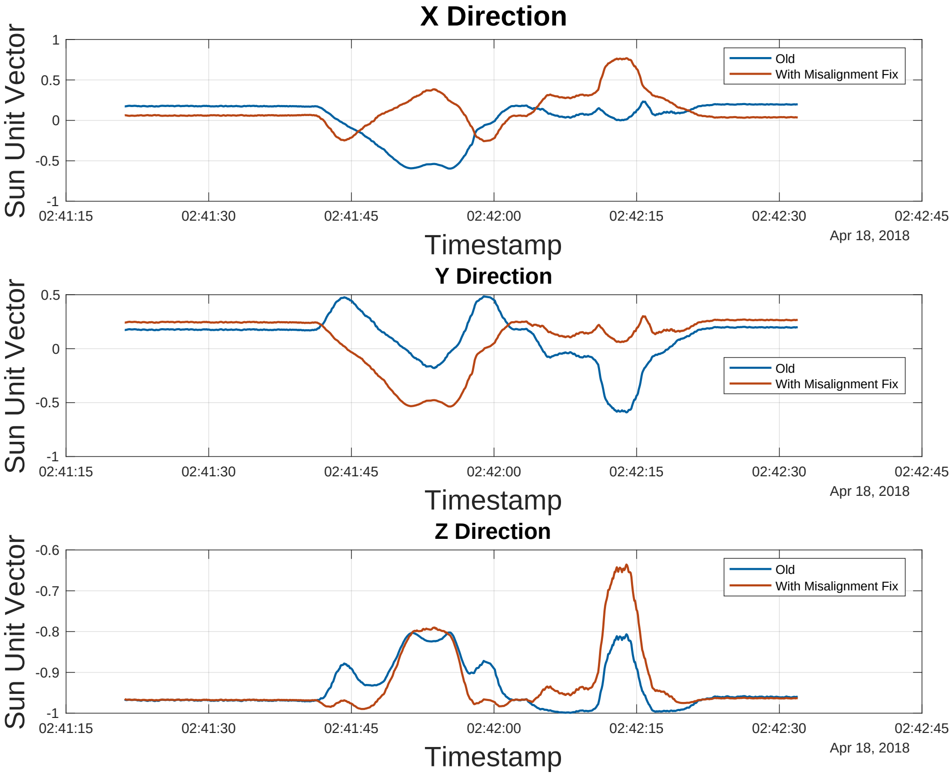


Figure 3.11: Times series data of sun sensor values with alignment rotation transformation compared to the non-aligned sun sensor values. The calibrated values are from the testbed study where it was rotated clockwise and counter-clockwise

3.4 CubeSat Attitude Determination

3.4.1 Three-Axis Attitude Determination (TRIAD)

The ADCS testbed body rotation in the inertial reference frame can be determined using the Three-axis attitude determination (TRIAD) algorithm as explained in this reference [8] given two non-parallel reference sensors, the magnetometer measurement vector and the sun sensor measurement vector. From the two non-parallel reference unit vectors $\hat{V}_1$ and $\hat{V}_2$ (i.e. the inertial reference) and their corresponding observation unit vectors $\hat{W}_1$ and $\hat{W}_2$ (e.g. magnetometer, sun sensor) an orthogonal matrix A or attitude rotation matrix can be determined which satisfies:

$$A\hat{V}_1 = \hat{W}_1 \quad A\hat{V}_2 = \hat{W}_2 \tag{3.29}$$

Because A is overdetermined begin by constructing two triads of manifestly orthonormal references (inertial) and observation (measured) vectors according to:

$$\hat{r}_1 = \frac{\hat{V}_1}{||\hat{V}_1||} \quad \hat{r}_2 = \frac{(\hat{V}_1 \times \hat{V}_2)}{||\hat{V}_1 \times \hat{V}_2||} \tag{3.30}$$

$$\hat{r}_3 = \frac{(\hat{r}_1 \times \hat{r}_2)}{||\hat{r}_1 \text{x} \hat{r}_2||} \tag{3.31}$$

$$\hat{s}_1 = \frac{\hat{W}_1}{||\hat{W}_1||} \quad \hat{s}_2 = \frac{(\hat{W}_1 \times \hat{W}_2)}{||\hat{W}_1 \times \hat{W}_2||} \tag{3.32}$$

$$\hat{s}_3 = \frac{(\hat{s}_1 \times \hat{s}_2)}{||\hat{s}_1 \times \hat{s}_2||} \tag{3.33}$$

where the vectors $\hat{r}_i$ are each vector component of the triad reference frame and $\hat{s}_i$ are each vector of the observation reference frame.

$$M_{ref} = \begin{bmatrix} \hat{r}_1 & \hat{r}_2 & \hat{r}_3 \end{bmatrix} \tag{3.34}$$

$$M_{obs} = \begin{bmatrix} \hat{s}_1 & \hat{s}_2 & \hat{s}_3 \end{bmatrix} \tag{3.35}$$

$$A = M_{obs} M_{ref}^T \tag{3.36}$$

A is the DCM of the ADCS system body reference frame in the inertial reference frame and extrapolating Euler Angles of testbed with respect to the inertial reference frame from A can be done using equations 1.9 - 1.11.

For the ADCS CubeSat testbed system we use the calibrated and aligned sun sensor and magnetometer values to determine A or the DCM for the testbed with respect to the reference unit vectors $\hat{W}_1$ and $\hat{W}_2$ being the nominal values when the testbed body frame is aligned with the inertial reference frame of the testing cage as seen in Fig. 2.1. The TRIAD algorithm attitude estimation case where the testbed was rotated clock-wise and then counter-clock wise on the airbearing was validated and verified by the AprilTag based ATS as seen in Fig. 3.12.

The TRIAD algorithm on-board the CubeSat testbed was accurate up to $\pm 5.3^o$ in roll and pitch and $\pm 5.5^o$ in yaw with respect to the ATS with EKF as seen in Fig. 3.12. The sources of these errors can be attributed to sensitivity of the TRIAD algorithm to the combined sensor noise of the sun sensor and magnetometer.

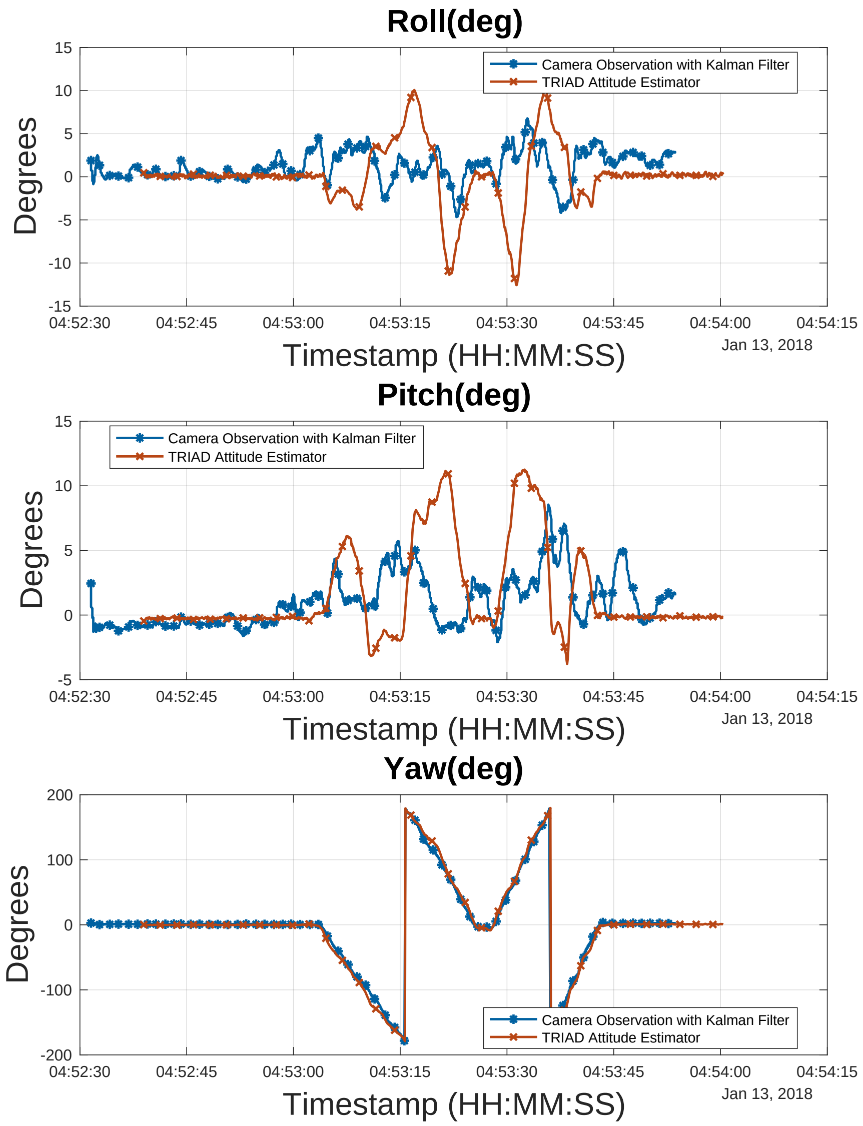


Figure 3.12: Time Series of ATS testbed observation with extended Kalman filter represented by a blue starred line side-by-side the with on-board estimated attitude by TRIAD represented by the red line with x's.

3.4.2 Complementary Filter (CF) with Bias Correction

A complementary filter (CF) based attitude estimation approach was implemented on-board the testbed using ($n \geq 2$) tri-axial vectors and the on-board gyroscope. The CF is an improvement to attitude estimation of the CubeSat testbed in comparison to the TRIAD implementation due to the integration of the gyro-measured body rates. The CF has bias correction which considers the rotation kinematics for a time varying R(t) ∈ SO(3) and with measurements from the on-board sun sensor, magnetometer and gyroscope.

The CF development for the ADCS testbed can be illustrated by examining the models from the three chosen sensors: gyroscope $\underline{\hat{\omega}} = \underline{\omega} + \mu_\omega$, magnetometer $\underline{\hat{m}} = \underline{m} + \mu$ and sun sensor $\underline{\hat{s}} = \underline{s} + \mu$ where μ_ω is predominantly low frequency noise and μ is high frequency noise. Choosing a pair of transfer functions $F_1(s) + F_2(s) = 1$ with $F_1(s)$ as the low pass filter and $F_2(s)$ as the high pass filter, the filtered signal $\hat{X}(s)$ is:

$$\begin{aligned}\hat{X}(s) &= X(s) + F_1(s)Y_\omega + F_2(s)Y_m \\ \hat{X}(s) &= X(s) + F_1(s)\mu_\omega + F_2(s)\mu \\ \hat{X}(s) &= X(s) + \frac{K_p}{K_p + s}\mu_\omega + \frac{s}{K_p + s}\mu\end{aligned}$$

Because the μ_ω is mostly due to the constant gyro null shift ($\underline{b}_o$) and bias drift ($\underline{b}(t)$) which are low frequency disturbances, a PI controller can be used to tune values of K_p and K_I to best estimate attitude.

$$\dot{\hat{x}} = \omega - Kp(y_{meas} - \hat{x}) - K_I(y_{meas} - \hat{x})\Delta t \tag{3.37}$$

The implementation of the PI controller for CF on S0(3) groups is as follows:

$$\dot{\hat{R}} = \hat{R}([\Omega + K_I \epsilon_{meas}]_\times) + K_p([\epsilon_{meas}]_x) \tag{3.38}$$

where ϵ_{meas}= Measurement vector (i.e. sun sensor or magnetometer) errors between body measurements and inertial. $\Omega = \underline{\omega}$(vector of body angular rates)

$$\dot{\hat{b}} = -K_I \omega_{meas} \tag{3.39}$$

$$\epsilon_{meas} = \sum_{i=1}^{n} k_i v_i x \hat{v}_i \tag{3.40}$$

and choose $k_i = 0$ such that M_o has three distinct eigenvalues $M_o = \sum_{i=1}^{n} k_i v_{o,i} v_{o,i}^T$, where $\hat{v_i} = \hat{R}^T v_{o,i}$.

The complementary filter implementation for attitude estimation on the ADCS testbed is a DCM based attitude determination algorithm as described in algorithm 3 ([22])

Before beginning the algorithm an initial bias vector B^+ (i.e. 3x1 vector of 0s) and an initial rotation $[R]^-$ (i.e. 3x3 identity) are guessed.

Algorithm 3: Complementary Filter Attitude Estimator

```
**Result**: Next Corrected Rotation Matrix Update $[R]^+$
**Function** CF($B^-$, $[R]^-$):
    $Ki_s = \frac{Kp_s}{10}$
    $Ki_m = \frac{Kp_m}{10}$
    $\underline{g}_{Withbias} = \underline{g} - B^-$
    $\underline{\omega}_s = [\underline{s}]_x([R^-]^T\underline{s_i})$
    $\underline{\omega}_m = [\underline{m}]_x([R^-]^T\underline{m_i})$
    $\underline{g}^+ = \underline{g}_{WithBias} + Kp_s\underline{w}_s + Kp_m\underline{w}_m$
    $r_x = (\underline{g}^+)_x$
    $g_{norm} = \frac{(\underline{g}^+\Delta t)}{||\underline{g}^+\Delta t||}$
    $Rexp_{meas} = I + [\text{sinc}(\frac{g_{norm}}{2\pi})]\cos[\frac{g_{norm}}{2}]^T r_x$
    $[R]^+ = [R]^-[Rexp_{meas}\Delta t)]$
    $\dot{b} = -Ki_s\underline{\omega}_s - Ki_m\underline{\omega}_m$
    $B^+ = B^- + \dot{b}$
    **return** $[R]^+$,$B^+$
```

where B^- is the previous gyro bias, $[R]^-$ is the previous corrected rotation matrix, $\underline{s}$ is the normalized sun sensor data, $\underline{s_i}$ is the normalized inertial sun direction, $\underline{m}$) is the normalized magnetometer sensor data, $\underline{m_i}$ is the normalized inertial magnetometer direction, $\underline{g}$ are the input angular rate values, B^+ is the next estimated gyro bias and $[R]^+$ is the next estimated corrected rotation matrix. Once algorithm 3 outputs B^+ and $[R]^+$ they become the inputs for the next iteration. Implementation of the

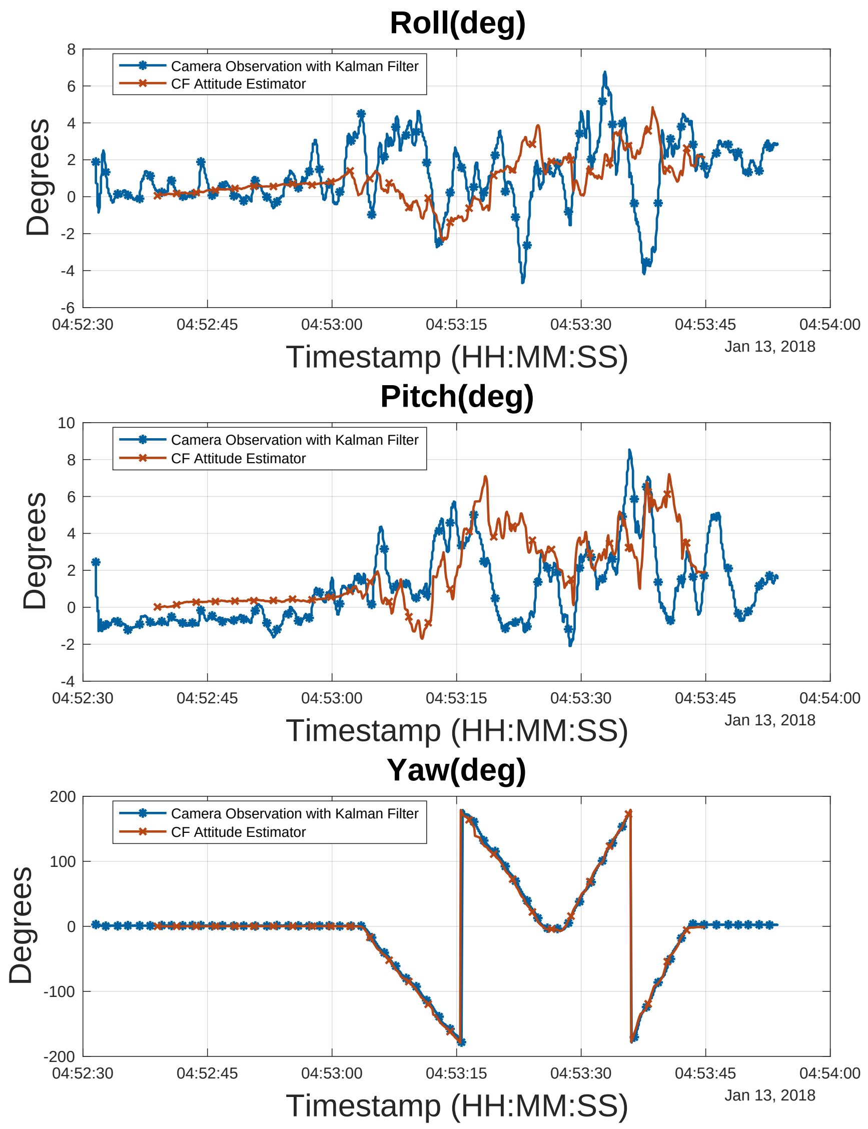


Figure 3.13: Time Series of ATS testbed observation with extended Kalman filter represented by a blue dotted line side-by-side the with on-board estimated attitude by the complementary filter represented by the red line with stars.

complementary filter algorithm 3 is illustrated in Fig. 3.13,

The CF algorithm 3 on-board the CubeSat testbed is accurate up to $\pm 2.1^o$ in roll and pitch and $\pm 2.9^o$ in yaw with respect to the ATS as seen in Fig. 3.13. The on-board attitude determination complementary filter is more accurate to the ATS system than the TRIAD attitude estimation algorithm by $|3.3°|$, $|3.3°|$ and $|2.6°|$ in roll, pitch and yaw, respectively. The difference can be seen more clearly in the side-by-side overlay plot of the implemented TRIAD attitude determination results and the complementary filter results with the ATS observation values in Fig. 3.14.

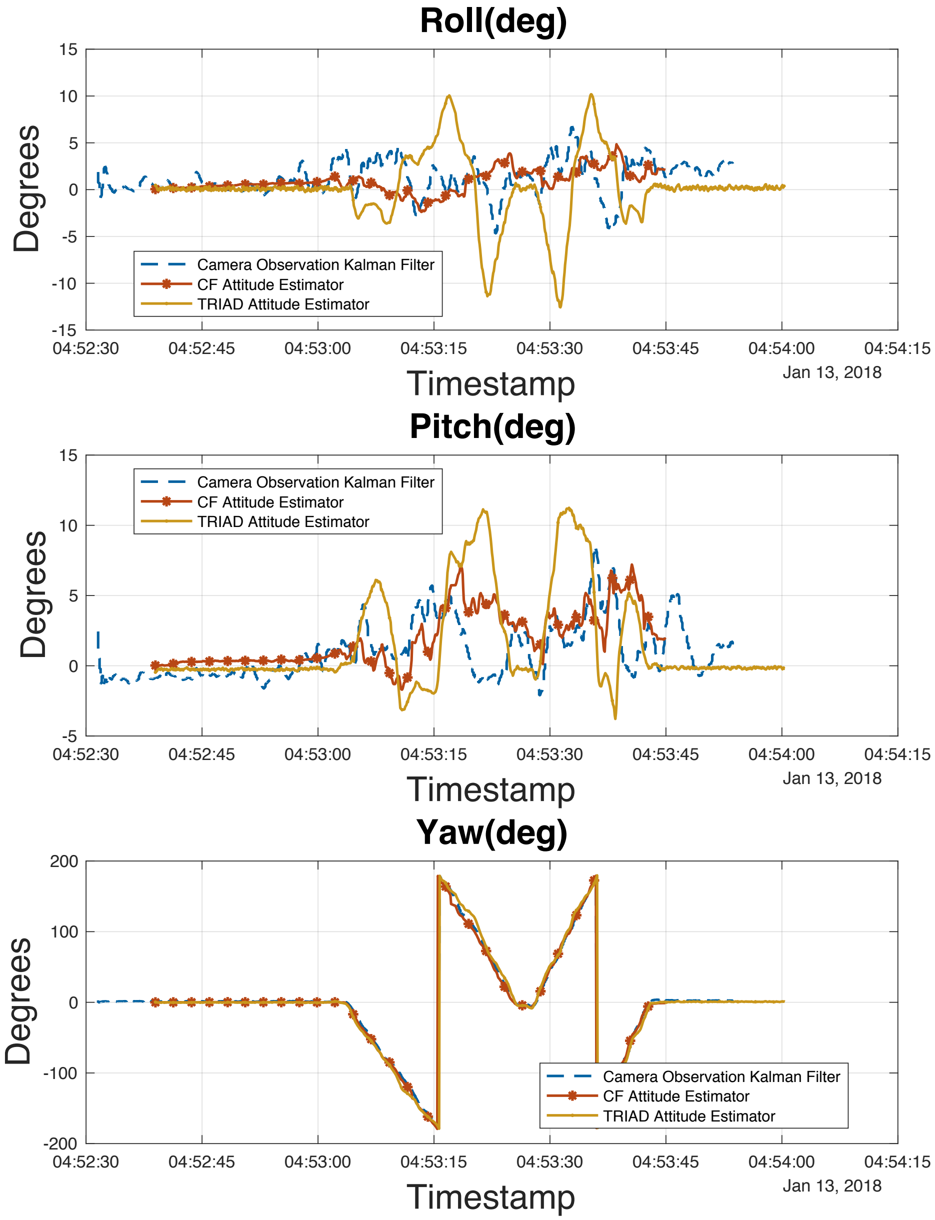


Figure 3.14: Time Series of ATS testbed observation with extended Kalman filter represented by a blue dotted line side-by-side the with on-board estimated attitude by the complementary filter represented by the red line with stars and TRIAD represented by the yellow line with dots.

Chapter 4

Conclusion

4.1 Summary

The increased popularity of very small spacecraft, frequently referred to as CubeSats, brings with it the possibility of low-cost satellite constellations or satellite swarms. Small satellite constellations are being tested and tried to create never before seen scientific experiments low Earth orbit (LEO) and beyond. The freedom of creativity provided by the boom in CubeSat projects have created a need for flexible, easily accessible facilities for iterative hardware and software testing for performance validation of CubeSat attitude determination and control system (ADCS) technologies. Generally the validation and iterative testing requirements for real-time satellite attitude estimation can accrue large costs (especially for clustered systems) due to time lost setting up a unique test environment for each mission and from the lack of standardized testing facilities to fine tune sensor integration and engineering test systems. This work helped progress the development of a low-cost, flexible, accurate and easy to use standardized CubeSat testing facility, the Generalized Nanosatellite Avionics Testbed (G-NAT) lab at NASA Ames Research Center.

During the progress of this work we have developed a laboratory to verify sensor and actuator performance requirements in a standardized testing facility complemented with a ultra low-cost tested AprilTag based ATS using a COTS high-definition web-cams attitude truth system [23],[18]. The research verified implementation of two attitude estimators TRIAD, and the complementary filter, using low-cost on-board sensors of the ADCS system (e.g. magnetometer, sun sensor and gyroscope). Additionally, the research study proved the viability of using the AprilTag based ATS as a tangible performance metric for ADCS performance validation.

Chapter 1 of this thesis described the motivation and test environment for the CubeSat facility, CubeSat dynamics and mathematical background for direction cosine matricies and non-linear rigid body rotation model. Additionally, the first chapter explained the motivation for a turnkey ATS using inexpensive cameras to complement the G-NAT facility.

Chapter 2 of this thesis described the product research for AprilTag, in house testing, calibration scheme development, and tracking system implementation using an EKF for robustness and performance improvement. The development of a ultra-cheap,portable and scalable ATS in the G-NAT laboratory shows a viable alternative system to track satellites, quadcopters, driverless vehicles and potentially unmanned underwater vehicles. Additionally, the contributions of this method can be used for many initial SLAM (Simultaneous Localization And Mapping) and robotic navigation research projects. The implementation of the EKF estimation overlaid on top of the camera attitude tracking system kept the accuracy of the tracking system sustained to within $\pm 1.30^o$, $\pm 1.30^o$ and $\pm 0.61^o$ for roll, pitch and yaw estimations, respectively. The ATS has met the initial testing facility performance well within the mission requirements and the visual markers are small enough that they can be readily integrated onto any relevant hardware testbed.

Chapter 3 described development of attitude estimation on-board the Cubesat ADCS testbed, sensor calibration, sensor alignment and validation with the AprilTag based ATS. The attitude determination tests have shown very briefly the accuracy of the two on-board algorithms with the ATS. The TRIAD algorithm for a variety of experiments has shown to be accurate to within $\pm 5.3^o$ in roll and pitch and $\pm 5.5^o$ in yaw on average using solely a magnetometer and a sun sensor. The CF algorithm on-board the CubeSat testbed was accurate up to $\pm 2.1^o$ in roll and pitch and $\pm 2.9^o$ in yaw with respect to the ATS. The on-board attitude determination complementary filter is more accurate to the ATS system than the TRIAD attitude estimation algorithm by 62.2%, 62.2% and 47.3% in roll, pitch and yaw, respectively. The objective of this lab is to advance the technology readiness level of ADCS and avionics technologies for small spacecraft missions under development by NASA and its partners in academia and industry. The functioning testbed environment, ATS and ADCS testbed system has proven its readiness for actual CubeSat mission testing.

4.2 Future Work

Each topic mentioned in this thesis from computer vision to attitude determination to sensor fusion to optimal control to CubeSat cluster synchronization can be explored further .

Specifically, in chapter 2 an area of further improvement is in the ATS attitude filters and architecture. The variation of camera attitude truth systems is growing rapidly and many current systems can be used to meet the various requirements mentioned in chapter 2. Immediate improvement can be made to the extended Kalman Filter used to estimate the camera observation information. Accuracy improvements can be made if one was to use a 2nd-order Kalman filter or Kalman Smoother. The camera attitude truth system architecture can be radically improved. For example one could implement the reflector based attitude truth system on the testbed facility with some image processing architectural changes [14]. The reflector based system can accurately record movement with mm resolution (which could mean $< 0.5^o$ accuracy) as long as multiple cameras are calibrated and setup the same object of interest and image data is processed in real-time. The trick would be passing the camera frames to the shared memory on the computer and processing object attitude in real-time and display a fraction of the camera frames for real-time streaming and real-time attitude display. No one has yet to setup an open source version that includes real-time video streaming in MATLAB [15]. Implementing this system completely eliminates the need for external AprilTag fiducial markers, but instead uses small circle markers (That can be magnitudes smaller than the AprilTag) or potentially geometric features of the ADCS testbed/Nanosatellite. Developing a turn-key, scalable, portable and OpenCV compatible version of this software could be a very exciting area of work. Additionally, the ATS system should be tested against more accurate commercial off-the-shelf attitude truth systems (e.g. Optitrack) to determine its true accuracy.

The ADCS system used on the testbed described in chapter 3 can be further improved specifically in the area of attitude estimators and multi-satellite synchronization. Significant attitude estimation improvements can be made exploring the various manifestations of Kalman filters and extended Kalman filters for improved on-board attitude determination accuracy and reach accuracy errors of $\pm 0.5^o$. Reference [2] describes an alternative attitude determination algorithm using a multiplicative extended Kalman filter (MEKF). Testing and implementing these two different attitude determination algorithms and getting verification from the ATS would be very useful future work from the G-NAT Lab.

In the G-NAT facility the main future work trajectory is to eventually test the case of two members of a CubeSat swarm communicating their attitude state with one another in real-time. The interaction between two fully functioning testbeds is essential for showing proof-of-concept of HIL and SIL satellite cluster synchronization. The synchronization can only be possible with at least one testbed having a functioning actuation system to have mechanical feedback to deliberately track a commanded attitude position. Specifically, a CubeSat testbed on one of the air bearings in the G-NAT lab will command the second testbed to perform a reorientation. The testbed performing the reorientation maneuver, will calculate its attitude on-board using the a attitude estimator (i.e. TRIAD or variations of the complementary filter). The finalized G-NAT test facility has great potential to be an incredible rapid prototyping and CubeSat validation facility to all further NASA Ames Research Center and collaborating institution CubeSat missions.

Bibliography

[1] J C Springmann and J W Cutler. *Satellite Attitude Determination with Low-Cost Sensors*. PhD thesis, University of Michgan, 2013.

[2] John C Springmann and James W Cutler. Photodiode Placement & Algorithms for CubeSat Attitude Determination. Technical report, University of Michgan, 2012.

[3] Space News Magazine. Cubesat swarms about to set space abuzz, May 19, 2016 (Accessed February 14, 2016).

[4] et al. Diaz-Aguado, M.F. Small class-d spacecraft thermal design, test, and analysis-pharmasat biological experiment. *Proceedings of the IEEE Aerospace Conference*, 1:1–9, 2009.

[5] M. M. Nehrenz and K. Shish. Operational considerations for a swarm of cubesat-class spacecraft. *Proceedings of AIAA SpaceOps Conference*, May 2014.

[6] D. Kemp M. Sorgenfrei, J. Liss. Verification of attitude determination and control capabilities for cubesat-class spacecraft. *AAS GNC*, Feb 2018.

[7] J. Miller L. Pei and Rothhaar P. Demonstration of the space launch system augmenting adaptive control algorithm on a quad-rotor. *Proceedings of AIAA SciTech Forum*, 2017.

[8] M. D. SHUSTER and S. D. OH. Three-axis attitude determination from vector observations. *J. Guid. Control. Dyn.*, 4(1):70–77, 1981.

[9] Peter Zipfel. *Modeling and Simulation of Aerospace Vehicle Dynamics, Second Edition*. American Institute of Aeronautics and Astronautics, Inc. Education Series, Gainsville, Florida, second edi edition, 2007.

[10] Jens Wittenburg. Kinematics: Theory and applications. *Kinemat. Theory Appl.*, pages 1–685, 2016.

[11] Daniel Eliahu, Gabriel H. Elkaim, and Renwick E Curry. Danny_Thesis.pdf. Master's thesis, University of California Santa Cruz, 2017.

[12] Marshall H Kaplan. *Modern Spacecraft Dynamics and Control*. John Wiley and Sons Inc., 1976.

[13] The Analytic Sciences Corporation Technical Staff. *Applied Optimal Estimation.* American Institute of Aeronautics and Astronautics, Inc., 1974.

[14] Tyson L. Hedrick. Software techniques for two- and three-dimensional kinematic measurements of biological and biomimetic systems. *Bioinspiration and Biomimetics*, 3(3), 2008.

[15] Matthew W Bundle, Michael O Powell, and Laurence J Ryan. Design and testing of a high-speed treadmill to measure ground reaction forces at the limit of human gait. *Med. Eng. Phys.*, 37(9):892–897, sep 2015.

[16] Andrew Zisserman Richard Hartley. *Multiple View Geometry in Computer Vision.* Cambridge Press, 2003.

[17] Edwin Olson. AprilTag: A robust and flexible visual fiducial system. In *Proc. - IEEE Int. Conf. Robot. Autom.*, pages 3400–3407, 2011.

[18] John Wang and Edwin Olson. AprilTag 2: Efficient and robust fiducial detection. *IEEE Int. Conf. Intell. Robot. Syst.*, 2016-Novem(1):4193–4198, 2016.

[19] Elkaim, Gabriel H. and Demoz Gebre-egziabher. *"Multisensor Attitude Estimation: Fundamental Concepts and Applications (Devices, Circuits, and Systems)"*. Ed. by Fourati Hassen et al. CRC Press, 2016.

[20] Eric Dorveaux, David Vissire, Alain Pierre Martin, and Nicolas Petit. Iterative calibration method for inertial and magnetic sensors. *Proc. IEEE Conf. Decis. Control*, pages 8296–8303, 2009.

[21] Gabriel Hugh Elkaim. Batch Misalignment Calibration of Multiple Three-Axis Sensors. *12th International Workshop on the Algorithmic Foundations of Robotics (WAFR 2016)*, 2016.

[22] Robert Mahony, Tarek Hamel, Pascal Morin, and Ezio Malis. Nonlinear complementary filters on the special linear group. *Int. J. Control*, 85(10):1557–1573, 2012.

[23] Sorgenfrei M. Nehrenz M. Edwards R. and Joshi S.S. Formulation of a small spacecraft avionics testbed. *Proceedings of the AAS Guidance, Navigation, and Control Conference*, February 2014.

[24] et al. Nissler, C. Evaluation and improvement of global pose estimation with multiple apriltags for industrial manipulators. *Proceedings of IEEE International Conference on Emerging Technologies in Factory Automation*, November 2016.

[25] Gene Franklin, J.David Powell, and Michael Workman. *Digital Control of Dynamic Systems.* Ellis-Kagle Press, 1997.

[26] J Carryer, R Ohline, and T Kenny. *Introduction to Mechatronic Design.* Prentice Hall, 2011.

[27] Gene Franklin and J.David Powell. *Feedback control of dynamic systems.* Pearson, 2009.