



USING PREDICTIVE RENDERING AS A VISION-AIDED TECHNIQUE  
FOR AUTONOMOUS AERIAL REFUELING

THESIS

Adam D. Weaver, Captain, USAF

AFIT/GE/ENG/09-45

DEPARTMENT OF THE AIR FORCE  
AIR UNIVERSITY

***AIR FORCE INSTITUTE OF TECHNOLOGY***

Wright-Patterson Air Force Base, Ohio

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

The views expressed in this thesis are those of the author and do not reflect the official policy or position of the United States Air Force, Department of Defense, or the United States Government.

USING PREDICTIVE RENDERING AS A VISION-AIDED TECHNIQUE  
FOR AUTONOMOUS AERIAL REFUELING

THESIS

Presented to the Faculty  
Department of Electrical and Computer Engineering  
Graduate School of Engineering and Management  
Air Force Institute of Technology  
Air University  
Air Education and Training Command  
In Partial Fulfillment of the Requirements for the  
Degree of Master of Science in Electrical Engineering

Adam D. Weaver, B.S.E.E.  
Captain, USAF

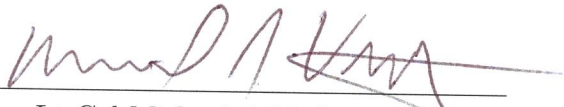
March 2009

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

USING PREDICTIVE RENDERING AS A VISION-AIDED TECHNIQUE  
FOR AUTONOMOUS AERIAL REFUELING

Adam D. Weaver, B.S.E.E.  
Captain, USAF


Approved:

  
Lt Col Michael J. Veth, PhD (Chairman)

5 MAR 09  
date

  
Dr. John F. Raquet (Member)

4 MAR 09  
date

  
Dr. Meir Pachter (Member)

5 March 2009  
date

## *Abstract*

This research effort seeks to characterize a vision-aided approach for an Unmanned Aerial System (UAS) to autonomously determine relative position to another aircraft in a formation, specifically to address the autonomous aerial refueling problem. A system consisting of a monocular digital camera coupled with inertial sensors onboard the UAS is analyzed for feasibility of using this vision-aided approach.

A three-dimensional rendering of the tanker aircraft is used to generate predicted images of the tanker as seen by the receiver aircraft. A rigorous error model is developed to model the relative dynamics between an INS-equipped receiver and the tanker aircraft. A thorough image processing analysis is performed to determine error observability between the predicted and true images using sum-squared difference and gradient techniques. To quantify the errors between the predicted and true images, an image update function is developed using perturbation techniques. Based on this residual measurement and the inertial/dynamics propagation, an Extended Kalman Filter (EKF) is used to predict the relative position and orientation of the tanker from the receiver aircraft.

The EKF is simulated through various formation positions during typical aerial refueling operations. Various grades of inertial sensors are simulated during different trajectories to analyze the system's ability to accurately and robustly track the relative position between the two aircraft.

## *Acknowledgements*

First, I would like to thank my advisor, Lt Col Mike Veth, for his guidance and advice throughout this research effort. Also, I would like to thank all my family back home for understanding the time commitment of completing this program.

Most of all, I owe a huge debt of gratitude to my wife. Her support and patience during this effort was a true blessing.

Adam D. Weaver

# *Table of Contents*

|   | Page  |
|---|-------|
| Abstract . . . . .  | iv    |
| Acknowledgements . . . . .                                      | v     |
| Table of Contents . . . . .                                     | vi    |
| List of Figures . . . . .                                       | viii  |
| List of Tables . . . . .  | xi    |
| List of Abbreviations . . . . .                                 | xii   |
| <br>I. Introduction . . . . .                                   | <br>1 |
| 1.1 Methods of Aerial Refueling . . . . .                       | 2     |
| 1.2 Research Objectives . . . . .                               | 3     |
| 1.3 Scope . . . . .   | 3     |
| 1.4 Overview . . . . .  | 4     |
| <br>II. Mathematical Background and Previous Research . . . . . | <br>6 |
| 2.1 General Mathematical Notation . . . . .                     | 6     |
| 2.2 Kalman Filtering . . . . .                                  | 6     |
| 2.2.1 Traditional Kalman Filter . . . . .                       | 7     |
| 2.2.2 Extended Kalman Filter . . . . .                          | 9     |
| 2.3 Reference Frames . . . . .                                  | 11    |
| 2.4 Coordinate Transformations . . . . .                        | 13    |
| 2.5 WGS-84 Reference System . . . . .                           | 14    |
| 2.6 Inertial Navigation Systems . . . . .                       | 14    |
| 2.6.1 Basic Functions of Components . . . . .                   | 15    |
| 2.6.2 Mechanization . . . . .                                   | 15    |
| 2.7 Optical Sensors . . . . .                                   | 17    |
| 2.8 Image Processing Techniques . . . . .                       | 18    |
| 2.8.1 Sum Squared Difference . . . . .                          | 20    |
| 2.8.2 Magnitude of Gradient . . . . .                           | 20    |
| 2.8.3 Magnitude of Gradient with Threshold . . . . .            | 21    |
| 2.9 Predictive Rendering Algorithm . . . . .                    | 22    |
| 2.9.1 Motivation . . . . .                                      | 22    |
| 2.9.2 Implementation . . . . .                                  | 24    |
| 2.10 Previous Research Efforts . . . . .                        | 27    |

|   | Page |
|---|------|
| 2.10.1 Ross/Spinelli Formation Flight Controller . . .                                  | 27   |
| 2.10.2 Spencer Optical Tracking for Relative Positioning                                | 27   |
| 2.10.3 Vision-Based Sensor and Navigation System . .                                    | 28   |
| 2.10.4 DARPA/NASA Dryden Probe and Drogue Refueling . . . . .                           | 29   |
| 2.10.5 Perugia University and West Virginia University                                  | 29   |
| 2.11 Summary . . . . .  | 30   |
| III. Algorithm Development and Implementation . . . . .                                 | 31   |
| 3.1 Modeling Aircraft Dynamics . . . . .  | 31   |
| 3.1.1 Wing Aircraft Dynamics . . . . .  | 31   |
| 3.1.2 Lead Aircraft Dynamics . . . . .  | 32   |
| 3.1.3 State-Space Model . . . . .   | 34   |
| 3.2 Image-Aided Filter Implementation . . . . .   | 37   |
| 3.2.1 Extended Kalman Filter Propagation . . . . .                                      | 37   |
| 3.2.2 Image Update . . . . .  | 37   |
| 3.3 Summary . . . . .   | 41   |
| IV. Algorithm Simulation and Analysis . . . . .   | 42   |
| 4.1 Analysis of Image Processing Techniques . . . . .                                   | 42   |
| 4.1.1 Sum-Squared Difference . . . . .  | 42   |
| 4.1.2 Magnitude of Gradient . . . . .   | 47   |
| 4.1.3 Magnitude of Gradient with Threshold . . . . .                                    | 52   |
| 4.1.4 Analysis . . . . .  | 55   |
| 4.2 Filter Analysis Using Generated Data . . . . .                                      | 57   |
| 4.2.1 Simple Trajectory at Pre-contact Position . . .                                   | 58   |
| 4.2.2 Simple Trajectory at Pre-contact Position with Initial Condition Errors . . . . . | 63   |
| 4.2.3 Simple Trajectory at the Contact Position . . .                                   | 67   |
| 4.3 Summary . . . . .   | 70   |
| V. Conclusions and Recommendations . . . . .  | 72   |
| 5.1 Research Conclusions . . . . .  | 72   |
| 5.2 Recommendations for Future Research . . . . .                                       | 73   |
| 5.2.1 Additional Sensor Analysis . . . . .  | 73   |
| 5.2.2 Modification for Real-Time Operation . . . . .                                    | 73   |
| Bibliography . . . . .  | 75   |



## *List of Figures*

| Figure |   | Page |
|--------|---|------|
| 1.1.   | Example of Probe Refueling . . . . .  | 2    |
| 1.2.   | Example of Boom Refueling . . . . .   | 3    |
| 2.1.   | Navigation Frames . . . . .   | 12   |
| 2.2.   | Diagram of Aircraft Body Axes . . . . .   | 13   |
| 2.3.   | Pinhole Camera Model . . . . .  | 17   |
| 2.4.   | Diagram of Aerial Refueling Formation Positions . . . . .   | 18   |
| 2.5.   | Sample Formation Position Images . . . . .  | 19   |
| 2.6.   | Example image after applying magnitude of gradient with thresh-<br>old. . . . .                                 | 22   |
| 2.7.   | Sample 3D Rendering of Tanker Aircraft. . . . .   | 25   |
| 2.8.   | LWIR Camera Mount during Flight Test Data Collection . . .  | 26   |
| 2.9.   | Rendered Image Prediction (Semi-Transparent) with True In-<br>frared Image. . . . .                             | 26   |
| 2.10.  | Aircraft in Formation Flight during Ross/Spinelli Flight Tests .  | 28   |
| 3.1.   | Example Error Distribution about the Best Image Match . . .   | 40   |
| 4.1.   | Perturbations using Sum-Squared Difference Approach . . . . .   | 43   |
| 4.2.   | Pre-contact position sum-squared difference approach with per-<br>turbations about $x$ axis and roll . . . . .  | 44   |
| 4.3.   | Pre-contact position sum-squared difference approach with per-<br>turbations about $x$ axis and pitch . . . . . | 44   |
| 4.4.   | Pre-contact position sum-squared difference approach with per-<br>turbations about $z$ axis and pitch . . . . . | 45   |
| 4.5.   | Contact position sum-squared difference approach with pertur-<br>bations about $x$ axis and roll . . . . .      | 46   |
| 4.6.   | Contact position sum-squared difference approach with pertur-<br>bations about $x$ axis and pitch . . . . .     | 46   |
| 4.7.   | Perturbations using Magnitude of Gradient Approach . . . . .  | 47   |

| Figure |  | Page |
|--------|--|------|
| 4.8.   | Pre-contact position magnitude of gradient approach with perturbations about $x$ axis and roll . . . . .                 | 48   |
| 4.9.   | Pre-contact position magnitude of gradient approach with perturbations about $y$ axis and yaw . . . . .                  | 49   |
| 4.10.  | Pre-contact position magnitude of gradient approach with perturbations about $y$ axis and roll . . . . .                 | 49   |
| 4.11.  | Pre-contact position magnitude of gradient approach with perturbations about $z$ axis and pitch . . . . .                | 50   |
| 4.12.  | Contact position magnitude of gradient approach with perturbations about $x$ axis and roll . . . . .                     | 51   |
| 4.13.  | Contact position magnitude of gradient approach with perturbations about $x$ axis and pitch . . . . .                    | 51   |
| 4.14.  | Perturbations using Magnitude of Gradient with Threshold Approach . . . . .  | 52   |
| 4.15.  | Pre-contact position magnitude of gradient with threshold approach with perturbations about $x$ axis and roll . . . . .  | 53   |
| 4.16.  | Pre-contact position magnitude of gradient with threshold approach with perturbations about $x$ axis and pitch . . . . . | 54   |
| 4.17.  | Contact position magnitude of gradient with threshold approach with perturbations about $x$ axis and roll . . . . .      | 54   |
| 4.18.  | Contact position magnitude of gradient with threshold approach with perturbations about $x$ axis and pitch . . . . .     | 55   |
| 4.19.  | Pre-contact Relative Position Error with No Image Updates . .  | 59   |
| 4.20.  | Pre-contact Relative Position Error with Navigation Grade IMU  | 60   |
| 4.21.  | Pre-contact Position Measurement Residuals with Navigation Grade IMU . . . . .   | 61   |
| 4.22.  | Pre-contact Attitude Measurement Residuals with Navigation Grade IMU . . . . .   | 61   |
| 4.23.  | Pre-contact Relative Position Error with No Image Updates . .  | 62   |
| 4.24.  | Pre-contact Relative Position Error with Tactical Grade IMU .  | 63   |

| Figure |  | Page |
|--------|--|------|
| 4.25.  | Pre-contact Position Measurement Residuals with Tactical Grade IMU . . . . .           | 64   |
| 4.26.  | Pre-contact Attitude Measurement Residuals with Tactical Grade IMU . . . . .           | 64   |
| 4.27.  | Pre-contact Relative Position Error with Incorrect Initial Conditions . . . . .        | 65   |
| 4.28.  | Pre-contact Position Measurement Residuals with Incorrect Initial Conditions . . . . . | 66   |
| 4.29.  | Pre-contact Attitude Measurement Residuals with Incorrect Initial Conditions . . . . . | 66   |
| 4.30.  | Contact Relative Position Error with Navigation Grade IMU . . . . .                    | 67   |
| 4.31.  | Contact Position Measurement Residuals with Navigation Grade IMU . . . . .             | 68   |
| 4.32.  | Contact Attitude Measurement Residuals with Navigation Grade IMU . . . . .             | 68   |
| 4.33.  | Contact Relative Position Error with Tactical Grade IMU . . . . .                      | 70   |
| 4.34.  | Contact Position Measurement Residuals with Tactical Grade IMU . . . . .               | 71   |
| 4.35.  | Contact Attitude Measurement Residuals with Tactical Grade IMU . . . . .               | 71   |

*List of Tables*

| Table |   | Page |
|-------|---|------|
| 4.1.  | IMU Performance Parameters . . . . .                  | 58   |
| 4.2.  | Initial State Uncertainties for Simulations . . . . . | 59   |

# *List of Abbreviations*

| Abbreviation |   | Page |
|--------------|---|------|
| UAS          | Unmanned Aerial System . . . . .                        | 1    |
| USAF         | United States Air Force . . . . .                       | 1    |
| UCAS         | Unmanned Combat Aerial System . . . . .                 | 1    |
| GPS          | Global Positioning System . . . . .                     | 1    |
| AFRL         | Air Force Research Laboratory . . . . .                 | 2    |
| AAR          | Automated Aerial Refueling . . . . .                    | 2    |
| INS          | Inertial Navigation System . . . . .                    | 3    |
| EKF          | Extended Kalman Filter . . . . .                        | 4    |
| WGN          | White Gaussian Noise . . . . .                          | 6    |
| 3D           | Three Dimensional . . . . .                             | 14   |
| HAE          | Height Above Ellipsoid . . . . .                        | 14   |
| VR           | Virtual Reality . . . . .                               | 24   |
| TSPI         | Time-Space Positioning Information . . . . .            | 25   |
| LWIR         | Long Wave Infrared . . . . .                            | 25   |
| IMU          | Inertial Measurement Unit . . . . .                     | 28   |
| LEDs         | Light Emitting Diodes . . . . .                         | 28   |
| DARPA        | Defense Advanced Research Projects Agency . . . . .     | 29   |
| NASA         | National Aeronautics and Space Administration . . . . . | 29   |

# USING PREDICTIVE RENDERING AS A VISION-AIDED TECHNIQUE FOR AUTONOMOUS AERIAL REFUELING

## I. Introduction

The advent of unmanned aerial systems (UAS) has revolutionized the way the United States Air Force (USAF) conducts operations. No longer are aerial platforms limited by human endurance, but rather the endurance of the aircraft itself. This limiting factor can be even further reduced with the use of automated in-flight UAS refueling. The first unmanned systems (RQ-4 Global Hawk and MQ-1 Predator) were used exclusively for reconnaissance and could loiter over a target for extreme lengths of time. The Predator soon transitioned into an armed reconnaissance role. Today's systems have evolved even further into hunter/killer systems such as the MQ-4 Reaper. One can easily see this trend will develop unmanned systems into a more complex, larger component of the battlefield in the years to come [12].

With the imminent arrival of Unmanned Combat Aerial Systems (UCAS), the need for tactical refueling will be vital for mission success. Designed for high performance and larger, more complicated weapon payloads, these systems will have both shorter range and loiter time than the current unmanned systems fielded today. These systems will likely be a critical component of the tactical battlefield, and commanders will require them to perform missions that are currently dominated by manned systems. Thus, the need for a robust, autonomous refueling technique is vital for force projection of unmanned systems and overall mission success in future USAF operations [2, 8, 12].

Many efforts have already shown success in solving this problem (articulated in detail in Chapter II), primarily using Global Positioning System (GPS) approaches [8, 13]. With GPS, possible satellite acquisition issues exist for the receiver aircraft due to the tanker blocking large areas of the sky above it [9]. Furthermore, relying on

GPS solutions in every combat environment could be detrimental when GPS denial and degradation exist due to jamming or sensor failure. The Air Force Research Laboratory's (AFRL) Automated Aerial Refueling (AAR) Program has an interest in determining how well an autonomous system can function in environments where GPS is either degraded or denied [20]. This research will directly help analyze this specific area of the AAR program.

### ***1.1 Methods of Aerial Refueling***

There are two standard methods for in-flight refueling operations. The first, employed by the United States Navy and Marine Corps, is known as the probe and drogue technique. For this method, the refueling aircraft has a refueling probe that makes contact with a drogue chute on a refueling hose that trails from the tanker aircraft. Figure 1.1 shows a Marine Corps F/A-18 Hornet receiving fuel from an Air Force KC-10 Extender.

The second major method of aerial refueling is the flying boom method used by the USAF. For this approach, the receiving aircraft maintains a formation position just below the tail of the tanker aircraft, and the boom operator onboard the tanker inserts the refueling boom into the refueling receptacle of the receiver. Figure 1.2 displays this technique.



Figure 1.1: An F/A-18 Hornet receiving fuel from a KC-10 Extender using the probe and drogue aerial refueling method



Figure 1.2: An F-22 Raptor receiving fuel from a KC-135 Stratotanker using the aerial boom refueling method

## ***1.2 Research Objectives***

By applying an image processing algorithm to images received from a monocular camera, this research will investigate if the receiver aircraft can accurately determine the relative attitude and range to the lead tanker aircraft. Fusing this image data with the onboard Inertial Navigation System (INS) data will provide a robust navigation technique for autonomous UAS formation flight that could potentially be independent of external sources such as GPS. The ultimate goal will be validating a robust, vision-aided technique for accurate relative positioning between the tanker and receiver aircraft.

## ***1.3 Scope***

This research will focus on achieving accurate relative position of a receiver aircraft to a KC-135R Stratotanker. As aforementioned, the baseline system will have a monocular camera as well as data from an on-board INS. The receiver aircraft is assumed to start in a position behind and below the tanker, using the boom method of aerial refueling. This will ensure the images are free from unwanted objects such



as ground clutter and other aircraft. This research will analyze potential differences in INS quality and consider various image-processing techniques to develop a robust, accurate approach.

#### **1.4 Overview**

This thesis will be organized as follows. Chapter II will highlight key mathematical background to include applicable coordinate systems used in the relative navigation algorithm as well as transformation techniques between them. Kalman filtering will be discussed in detail, to include the basic Kalman Filter as well as the specific implementation used in this research, the Extended Kalman Filter (EKF). Then, fundamentals of an INS will be discussed as well as example mechanization equations. Also, some fundamental concepts of imaging sensors are presented. The chapter concludes by analyzing previous work concerning the autonomous aerial refueling problem, and how this research seeks to complement those previous efforts and exploit any lessons learned.

Chapter III presents the development of a predictive rendering environment to generate predicted images of the tanker aircraft as seen by the receiver. Several image-processing techniques (sum-squared difference, magnitude of gradient, and magnitude of gradient with threshold) are analyzed to determine their suitability for detecting errors between the true and predicted images. A relative dynamics model is developed to simulate the wing and lead aircraft flight characteristics during the refueling operation. A discussion of the EKF implementation, including the use of image updates, concludes the chapter.

Chapter IV analyzes the results of various simulations using the filter described in Chapter III. Several trajectories are analyzed with the receiver and tanker aircraft in different positions of typical refueling formations (e.g., pre-contact and contact). A presentation of filter performance with various grades of inertial sensors and initial errors is also presented based on the simulation results.

Finally, Chapter V will highlight the overall conclusions of the image-aided filter for autonomous refueling, analyzing the overall performance of the technique. An overall analysis of the research accomplishments and recommendations for future research conclude the chapter.

## II. Mathematical Background and Previous Research

This chapter describes the key mathematical and technical background information necessary to understand the algorithm development of Chapter III. These key concepts are highlighted in the beginning sections of this chapter and include: Kalman filtering, coordinate systems and transformations, INS description and mechanization, optical sensors, and image processing. The last section of the chapter highlights the current state of autonomous aerial refueling research.

### 2.1 General Mathematical Notation

Before developing mathematical equations, a brief description of the general notation is in order. For this document, scalar quantities will be identified by lower-case variables (e.g.,  $x$ ). Vector quantities will be identified by bold, lower-case variables (e.g.,  $\mathbf{x}$ ), and matrices will be identified by bold, upper-case variables (e.g.,  $\mathbf{X}$ ).

### 2.2 Kalman Filtering

The Kalman Filter is an optimal estimation technique frequently employed in navigation systems that uses statistical techniques to estimate the navigation state. There are many varieties of this technique, but the two discussed here are the traditional Kalman Filter and the Extended Kalman Filter. Before constructing this optimal estimator, a mathematical model must be constructed to approximate the dynamics of the system. Using a state-space format, the state vector of interest,  $\mathbf{x}$ , changes over time by the following expression

$$\dot{\mathbf{x}}(t) = \mathbf{F}(t)\mathbf{x}(t) + \mathbf{B}(t)\mathbf{u}(t) + \mathbf{G}(t)\mathbf{w}(t) \quad (2.1)$$

where  $\mathbf{F}(t)$  is the state dynamics matrix,  $\mathbf{B}(t)$  is the input matrix,  $\mathbf{u}(t)$  is the vector of deterministic inputs,  $\mathbf{G}(t)$  is the noise distribution matrix, and  $\mathbf{w}(t)$  is the noise vector. The values in the noise vector are typically assumed to be white, Gaussian noise (WGN) sources. Similarly, discrete-time measurements are used to update the

filter and are typically of the form:

$$\mathbf{z}(t_k) = \mathbf{H}(t_k)\mathbf{x}(t_k) + \mathbf{v}(t_k) \quad (2.2)$$

where  $\mathbf{z}(t_k)$  are the discrete time measurements,  $\mathbf{H}(t_k)$  is the measurement observability matrix, and  $\mathbf{v}(t_k)$  is the discretized measurement noise vector.

*2.2.1 Traditional Kalman Filter.* Now, with a stochastic model of the system, the traditional Kalman Filter can be constructed. This estimator tracks a state estimate ( $\hat{\mathbf{x}}$ ) and a state error covariance ( $\mathbf{P}$ ) through an alternating series of time-propagation and measurement updates. These and other variables used in the filter will often have a subscript and superscript notation. The subscript notation indicates at which time epoch the variable is evaluated. The superscript indicates whether this is immediately before or after a measurement update occurs. For example, the notation of  $\mathbf{P}_k^-$  indicates the *a priori* state error covariance estimate just before a measurement is taken at some time epoch  $k$ . The notation of  $\mathbf{P}_k^+$  indicates the *a posteriori* state error covariance estimate just after a measurement is taken at some time epoch  $k$ .

To construct the filter, the first step is to write the discrete-time stochastic difference equation from the continuous time model shown above. Before writing this equation directly, there are a few specific calculations that must be made. The needed values are  $\mathbf{B}_k$ ,  $\Phi(\Delta t)$ , and  $\mathbf{Q}_k$ .  $\mathbf{B}_k$  is the discretized version of the input matrix  $\mathbf{B}(\mathbf{t})$ ,  $\Phi(\Delta t)$  is the discrete time state transition matrix over the specific time interval, and  $\mathbf{Q}_k$  is the discretized process noise. These matrices are calculated as follows [11]:

$$\mathbf{B}_k = \int_{t_k}^{t_{k+1}} \Phi(t_{k+1} - \tau) \mathbf{B} \mathbf{u}(\tau) d\tau \quad (2.3)$$

where:

$$\Phi(\Delta t) = e^{\mathbf{F}\Delta t} \quad (2.4)$$

$\mathbf{Q}_k$  is solved using the Van Loan calculation, which is performed as follows:

$$\mathbf{A}^* = \begin{bmatrix} -\mathbf{F} & \mathbf{G}\mathbf{Q}\mathbf{G}^T \\ \mathbf{0} & \mathbf{F}^T \end{bmatrix} \quad (2.5)$$

$$\mathbf{B}^* = e^{\mathbf{A}^*} = \begin{bmatrix} \cdots & \Phi^{-1}\mathbf{Q}_k \\ 0 & \Phi^T \end{bmatrix} = \begin{bmatrix} \mathbf{B}_{11} & \mathbf{B}_{12} \\ \mathbf{B}_{21} & \mathbf{B}_{22} \end{bmatrix} \quad (2.6)$$

$$\mathbf{Q}_k = \mathbf{B}_{22}^T \mathbf{B}_{12} \quad (2.7)$$

Now, the stochastic difference equation can be formed as seen in Equation 2.8:

$$\mathbf{x}_{k+1} = \Phi_k \mathbf{x}_k + \mathbf{B}_k \mathbf{u}_k + \mathbf{w}_k \quad (2.8)$$

Now all the necessary elements exist to write the equations for executing the Kalman filter. The first function of the Kalman filter, time propagation, is accomplished with the following two calculations for each time step [11]:

$$\hat{\mathbf{x}}_k^- = \Phi_{k-1} \hat{\mathbf{x}}_{k-1}^+ + \mathbf{B}_{k-1} \mathbf{u}_{k-1} \quad (2.9)$$

$$\mathbf{P}_k^- = \Phi_{k-1} \mathbf{P}_{k-1}^+ \Phi_{k-1}^T + \mathbf{Q}_{k-1} \quad (2.10)$$

Then, at each time epoch  $k$  where a measurement is available, the following measurement update calculations are made:

$$\mathbf{K}_k = \mathbf{P}_k^- \mathbf{H}_k^T [\mathbf{H}_k \mathbf{P}_k^- \mathbf{H}_k^T + \mathbf{R}_k]^{-1} \quad (2.11)$$

$$\hat{\mathbf{x}}_k^+ = \hat{\mathbf{x}}_k^- + \mathbf{K}_k (\mathbf{z}_k - \mathbf{H}_k \hat{\mathbf{x}}_k^-) \quad (2.12)$$

$$\mathbf{P}_k^+ = \mathbf{P}_k^- - \mathbf{K}_k \mathbf{H}_k \mathbf{P}_k^- \quad (2.13)$$

The matrix  $\mathbf{K}_k$  is known as the Kalman gain and the matrix  $\mathbf{R}_k$  is the measurement uncertainty. This calculation process is repeated for the duration of filter execution [11].

*2.2.2 Extended Kalman Filter.* In many system models, the system dynamics and/or measurements are nonlinear functions, and cannot be simply written in terms of matrix multiplications as seen in Equations 2.1 and 2.2. The nonlinear dynamics and measurements are typically written in the following form [10]:

$$\dot{\mathbf{x}}(t) = \mathbf{f}[\mathbf{x}(t), \mathbf{u}(t), t] + \mathbf{G}(t)\mathbf{w}(t) \quad (2.14)$$

$$\mathbf{z}(t_i) = \mathbf{h}[\mathbf{x}(t_i)] + \mathbf{v}(t_i) \quad (2.15)$$

The Extended Kalman Filter (EKF) is one mechanism to account for these nonlinear effects. The EKF is an iterative algorithm where the filter is relinearized about each state estimate,  $\hat{\mathbf{x}}(t_i^+)$ , once it is computed. This results in creating a new, updated state trajectory in order to better estimate the next time propagation. This process should minimize the difference between the nominal and true trajectories, and hence improve the overall model [10].

In the general case, at a time  $t_i$ , immediately after a measurement is incorporated and a relinearization about the new state estimate is calculated, the best estimate of  $\delta\mathbf{x}(t_i)$  is  $\delta\hat{\mathbf{x}}(t_i^+) = 0$ . This is due to the relinearization just accomplished about the best state estimate [10]. Since the current estimate is zero, propagating it to the next sample time  $t_{i+1}$  will also yield zero. Using this knowledge, the measurement update for the relinearized filter is:

$$\delta\hat{\mathbf{x}}(t_{i+1}^+) = \mathbf{K}(t_{i+1})(\mathbf{z}_{i+1} - h[\mathbf{x}_n(t_{i+1}/t_i), t_{i+1}]) \quad (2.16)$$

The notation  $\mathbf{x}_n(t_{i+1}/t_i)$  is used to define the nominal state over the interval  $t_i$  to  $t_{i+1}$ .

Now to find the optimal estimate of the full state  $\hat{\mathbf{x}}(t/t_i)$ , simply combine the nominal with the perturbation estimate [10]:

$$\hat{\mathbf{x}}(t/t_i) = \mathbf{x}_n(t/t_i) + \delta\hat{\mathbf{x}}(t/t_i) \quad (2.17)$$

The equations that follow in this section form the iterative algorithm for executing the EKF. With the given dynamics model in the previous section, the measurement update step at time  $t_i$  incorporates the current measurement,  $\mathbf{z}_i$ , through the following [10]:

$$\mathbf{K}(t_i) = \mathbf{P}(t_i^-) \mathbf{H}^T[t_i; \hat{\mathbf{x}}(t_i^-)] (\mathbf{H}[t_i; \hat{\mathbf{x}}(t_i^-)] \mathbf{P}(t_i^-) \mathbf{H}^T[t_i; \hat{\mathbf{x}}(t_i^-)] + \mathbf{R}(t_i))^{-1} \quad (2.18)$$

$$\hat{\mathbf{x}}(t_i^+) = \hat{\mathbf{x}}(t_i^-) + \mathbf{K}(t_i)(\mathbf{z}_i - h[\hat{\mathbf{x}}(t_i^-), t_i]) \quad (2.19)$$

$$\mathbf{P}(t_i^+) = \mathbf{P}(t_i^-) - \mathbf{K}(t_i) \mathbf{H}[t_i; \hat{\mathbf{x}}(t_i^-)] \mathbf{P}(t_i^-) \quad (2.20)$$

where  $\mathbf{H}[t_i; \hat{\mathbf{x}}(t_i^-)]$  is defined as the partial derivative matrix [10]:

$$\mathbf{H}[t_i; \hat{\mathbf{x}}(t_i^-)] = \left. \frac{\partial h[\mathbf{x}, t_i]}{\partial \mathbf{x}} \right|_{\mathbf{x}=\hat{\mathbf{x}}(t_i^-)} \quad (2.21)$$

The estimate is propagated to the next sample time  $t_{i+1}$  by integrating the following differential equations:

$$\dot{\hat{\mathbf{x}}}(t/t_i) = f[\hat{\mathbf{x}}(t/t_i), \mathbf{u}(t), t] \quad (2.22)$$

$$\dot{\mathbf{P}}(t/t_i) = \mathbf{F}[t; \hat{\mathbf{x}}(t/t_i)] \mathbf{P}(t/t_i) + \mathbf{P}(t/t_i) \mathbf{F}^T[t; \hat{\mathbf{x}}(t/t_i)] + \mathbf{G}(t) \mathbf{Q}(t) \mathbf{G}^T(t) \quad (2.23)$$

The initial conditions for these integrations are based on the measurement update values in Equations 2.19 and 2.20.

$$\hat{\mathbf{x}}(t_i/t_i) = \hat{\mathbf{x}}(t_i^+) \quad (2.24)$$

$$\mathbf{P}(t_i/t_i) = \mathbf{P}(t_i^+) \quad (2.25)$$

The matrix  $\mathbf{F}[t; \hat{\mathbf{x}}(t/t_i)]$  from Equation 2.23 is defined as [10]:

$$\mathbf{F}[t; \hat{\mathbf{x}}(t/t_i)] = \left. \frac{\partial f[\mathbf{x}, \mathbf{u}(t), t]}{\partial \mathbf{x}} \right|_{\mathbf{x}=\hat{\mathbf{x}}(t/t_i)} \quad (2.26)$$

The linear stochastic difference equations can then be written in the form:

$$\delta \mathbf{x}(t_{i+1}) = \Phi(t_{i+1}, t_i) \delta \mathbf{x}(t_i) + \mathbf{w}_d(t_i) \quad (2.27)$$

$$\mathbf{P}(t_i)^- = \Phi(t_{i+1}, t_i) \mathbf{P}(t_i) \Phi^T(t_{i+1}, t_i) + \mathbf{Q}_d(t_i) \quad (2.28)$$

After propagating to the next sample time, the new values are defined as:

$$\hat{\mathbf{x}}(t_{i+1}^-) = \hat{\mathbf{x}}(t_{i+1}/t_i) \quad (2.29)$$

$$\mathbf{P}(t_{i+1}^-) = \mathbf{P}(t_{i+1}/t_i) \quad (2.30)$$

These propagate and update steps are iterated throughout the execution of the EKF [10].

### 2.3 Reference Frames

Before delving into the details of inertial navigation systems, one must first understand the reference frame about which navigation systems operate and define position. This research will primarily utilize frames presented in [17]. All are orthogonal and follow the right-hand convention, and are defined in the paragraphs that follow. A graphical representation from [17] is shown in Figure 2.1.

The inertial frame (*I-frame*) has fixed axes that do not rotate in space in relation to the fixed stars. This frame is where Newton's laws of motion are valid.

The Earth-fixed inertial frame (*i-frame*) has origin at the center of the Earth with the  $z$  axis aligned with the axis of rotation, and  $x$  and  $y$  axes in the equatorial plane. The axes do not rotate in relation to the fixed stars.

The Earth-centered Earth-fixed frame (*e-frame*) also has origin at the center of the Earth. The  $z$  axis is aligned with the axis of rotation, but the  $x$  is aligned with the intersection of the equatorial plane and the prime meridian (Greenwich). The



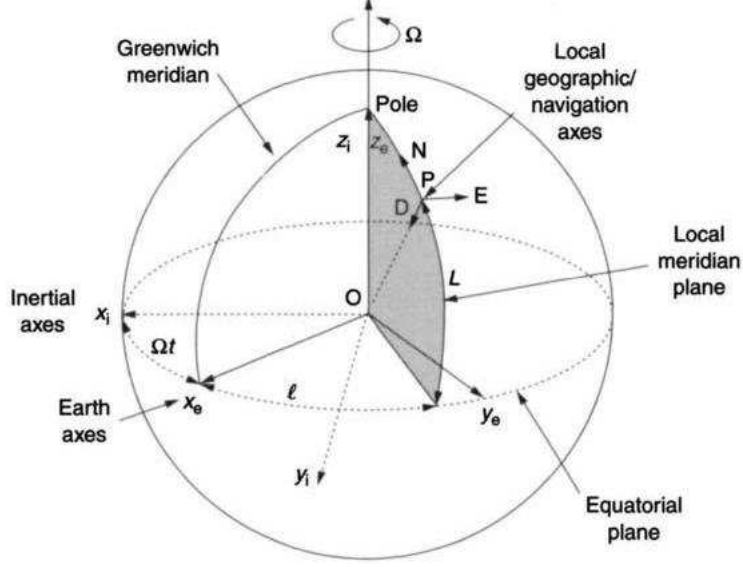


Figure 2.1: Diagram of Navigation Reference Frames [17]

$y$  axis is orthogonal based on the right-hand convention. The  $e$ -frame rotates with respect to the inertial frame.

The navigation frame ( $n$ -frame) has its origin at the location of the inertial navigation system (INS) or a fixed point on the surface of the Earth. The  $x$  and  $y$  axes are parallel to the surface of the Earth (aligned with North and East respectively), and the  $z$  axis points in the direction of local gravity (down). This is the North-East-Down (NED) convention.

The body frame ( $b$ -frame) has its origin at the center of mass of a vehicle. The  $x$ ,  $y$ , and  $z$  axes are aligned with the roll ( $\phi$ ), pitch ( $\theta$ ), and yaw ( $\psi$ ) axes, respectively. The three possible translations about the body axes ( $x$ ,  $y$ , and  $z$ ) and the three rotations (roll, pitch, and yaw) are the six degrees of freedom for the aircraft. Figure 2.2 illustrates a typical aircraft  $b$ -frame. Since this research will concern two aircraft flying in formation, the body frames of each will be designated based on their formation position. Coordinates relative to the lead aircraft (tanker) body frame will be denoted with an  $L$  (e.g.,  $\mathbf{p}^L$ ). Coordinates relative to the wing aircraft (receiver) body frame will be denoted with a  $W$  (e.g.,  $\mathbf{p}^W$ ).

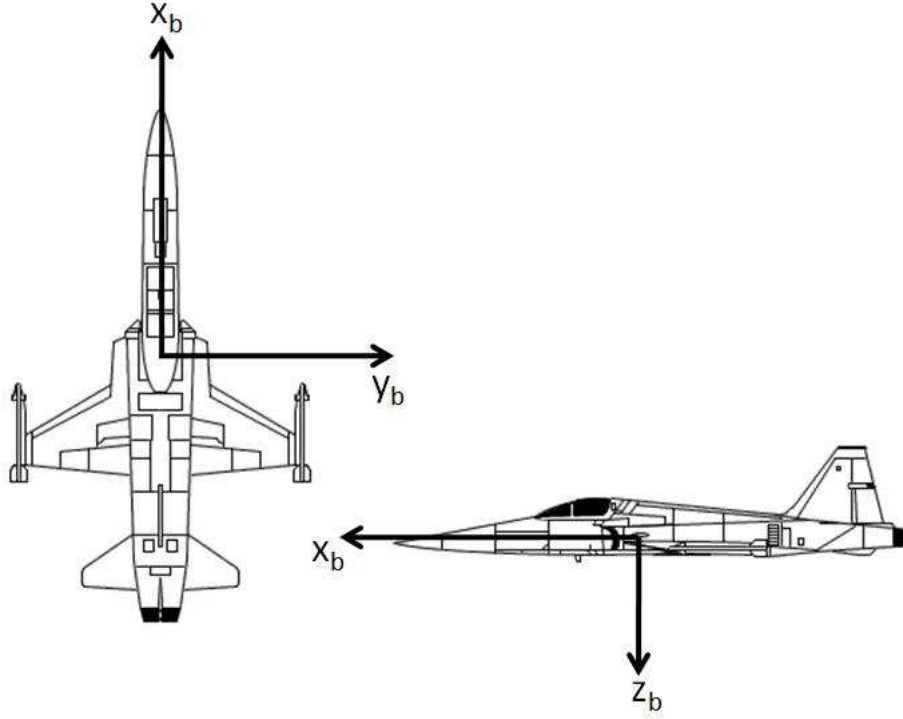


Figure 2.2: Example aircraft body frame illustration

## 2.4 *Coordinate Transformations*

With the large number of available reference frames, a vector will often need to be resolved from one frame into another frame. This brings rise to the coordinate transformation, which is often accomplished by use of the direction cosine matrix (DCM). The DCM from frame  $a$  to frame  $b$  is denoted by  $\mathbf{C}_a^b$ . As shown in [17], this  $3 \times 3$  matrix is the product of every unit basis vector in frame  $a$  with every unit basis vector in frame  $b$ . Each element in the DCM,  $C_{ij}$ , is the cosine of the angle between the  $i$  axis in the  $b$  frame with the  $j$  axis in the  $a$  frame.

A vector in basis  $b$  can easily be calculated by pre-multiplying the vector of basis  $a$  by the DCM as shown below:

$$\mathbf{r}^b = \mathbf{C}_a^b \mathbf{r}^a \quad (2.31)$$

## 2.5 WGS-84 Reference System

The Earth is a highly irregular body and various methods have been employed to model the Earth's shape. Though similar to a sphere, the Earth bulges at the equator and is more flattened at the poles. For precise navigation, modeling the Earth as a sphere leads to errors. The ellipsoid has been adopted as the most common datum to model the Earth's shape. A geodetic datum defines a three dimensional (3D) coordinate system to identify a specific location in terrestrial space. These dimensions consist of geodetic latitude and longitude and height (above the ellipsoid) [17].

In [1], The World Geodetic System Committee created a datum which approximates the shape of the Earth over the entire globe. Created in 1984, this datum is known as WGS-84. For mapping, charting, geopositioning, and navigation, WGS-84 is the most widely used global geodetic reference. With the origin at the center of the Earth, the  $z$  axis aligns with the axis of rotation and the  $x$  and  $y$  axes align with the equatorial plane. This is coincident with the *e-frame* described in the previous section. The  $x$  axis points toward the Zero Meridian, with the  $y$  axis orthogonal to it. From Figure 2.1, longitude ( $l$ ) is the angle between the local meridian plane and the Greenwich meridian. Latitude ( $L$ ) is the angle, in the local meridian plane, between the local navigation axes and the Equator. The mathematical model which approximates the shape of the Earth is known as the ellipsoid. Position coordinates in the WGS-84 reference frame are given in latitude, longitude, and height above the ellipsoid (HAE).

## 2.6 Inertial Navigation Systems

This section describes the basic aspects of an inertial navigation system (INS). From [21], an INS is composed of accelerometers and gyroscopes to sense motion, as well as a computer. With an initial set of conditions provided by the user, the INS is able to determine motion and relative position based on integrations of the raw measurements from the devices. These raw measurements are used to determine change in velocity, change in attitude, and change in geographic position.

*2.6.1 Basic Functions of Components.* INS systems are standalone, meaning they do not require input from any outside source. Based on knowledge of initial conditions, the system is able to track changes in velocity and position without the aid of an external reference. There are two different physical configurations for most INS instruments, gimbaled and strapdown. Gimbaled instruments are mounted to a platform that is connected to gimbals, which allow the platform to stay in a fixed orientation independent of the attitude of the vehicle. Based on inputs from the sensors, torque motors rotate the platform to counteract any motion by the vehicle. These systems are highly accurate, though they require a high-level of maintenance due to complex moving parts. Strapdown systems, just as the name suggests, are fixed to the body frame of the vehicle. Typically, one gyroscope and one accelerometer are aligned with each axis to measure each axis of rotation and each translational acceleration. Velocity and position are calculated by integrating the raw data from the gyroscopes and accelerometers [21].

Gyroscopes are able to accurately measure the rotational accelerations of the vehicle body with respect to inertial space. Accelerometers, however, measure specific force, which is the sum of gravity and the acceleration of the vehicle with respect to inertial space. Thus, they are unable to separate the ever-present effects of gravity. Hence, an INS computer must subtract the local gravitational field intensity to resolve true acceleration relative to inertial space [17].

The effect of the Earth's rotation must also be accounted for in INS position calculations. The Coriolis equations factor in the effects of the rotating Earth on the moving vehicle. Though not discussed in detail here, they are an important factor for INS design [17].

*2.6.2 Mechanization.* Now that the main components of the INS section are introduced, the mechanization for the INS can be discussed.

As discussed in [6], the INS navigation equations are used to describe how the position, velocity, and attitude are updated over time. Using the angular rate

measurement,  $\omega_{ib}^b$ , the attitude solution is updated in the body to Earth-frame DCM,  $\mathbf{C}_b^e$ , through the following calculation

$$\begin{aligned}\dot{\mathbf{C}}_b^e &= \mathbf{C}_b^e \boldsymbol{\Omega}_{eb}^b \\ &= \mathbf{C}_b^e \boldsymbol{\Omega}_{ib}^b - \boldsymbol{\Omega}_{ie}^e \mathbf{C}_b^e\end{aligned}\tag{2.32}$$

The terms  $\boldsymbol{\Omega}_{ie}^e$  and  $\boldsymbol{\Omega}_{ib}^b$  are the skew symmetric matrices of the Earth-rotation vector and the IMU's angular-rate measurement, respectively.

To update velocity, one must note that the centrifugal and Coriolis components must be included to account for the rotation of the Earth-frame axes with respect to inertial space. The velocity update is computed as:

$$\dot{\mathbf{v}}_{eb}^e = -\boldsymbol{\Omega}_{ie}^e \boldsymbol{\Omega}_{ie}^e \mathbf{p}_{eb}^e - 2\boldsymbol{\Omega}_{ie}^e \mathbf{v}_{eb}^e + \mathbf{a}_{ib}^e\tag{2.33}$$

The acceleration,  $\mathbf{a}_{ib}^e$ , is the sum of both the measured specific force ( $\mathbf{f}_{ib}^e$ ) and the gravitational force ( $\gamma_{ib}^e$ ). Also, one can define the term,  $\mathbf{g}_b^e$ , as the sum of the centrifugal and gravitational accelerations:

$$\mathbf{g}_b^e = \gamma_{ib}^e - \boldsymbol{\Omega}_{ie}^e \boldsymbol{\Omega}_{ie}^e \mathbf{p}_{eb}^e\tag{2.34}$$

Thus, making these two substitutions to Equation 2.33 yields:

$$\dot{\mathbf{v}}_{eb}^e = \mathbf{f}_{ib}^e + \mathbf{g}_b^e(\mathbf{p}_{eb}^e) - 2\boldsymbol{\Omega}_{ie}^e \mathbf{v}_{eb}^e\tag{2.35}$$

The position change is simply equal to the velocity:

$$\dot{\mathbf{p}}_{eb}^e = \mathbf{v}_{eb}^e\tag{2.36}$$

## 2.7 Optical Sensors

This section describes some fundamental concepts regarding camera models. For this research, a pinhole camera model was assumed for the digital camera onboard the wing (receiver) aircraft. This is consistent with the flight data used for image processing analysis, which will be discussed in detail in Chapters III and IV.

Figure 2.3 shows a typical pinhole camera model. The image plane is projected one focal length ( $f$ ) in front of the lens. As shown in the figure, the coordinate frame for the camera is right-handed, with the  $x$  and  $y$  axes pointing down and right, respectively. The  $z$  axis points perpendicular to the camera lens. For this research, there is no need to track specific pixels in any images (i.e., the location of the projection on the image plane).

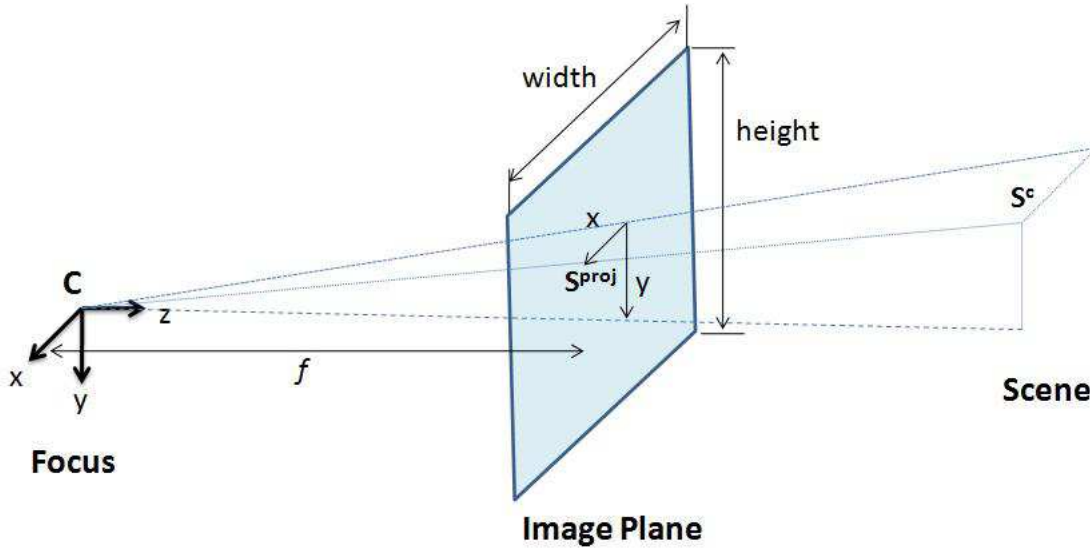


Figure 2.3: Simple Camera Projection Model. The virtual image plane is placed in between the focal point and the scene, thus the image is not inverted as typical in optical models.

## 2.8 Image Processing Techniques

During a refueling operation, the digital camera onboard the UAS will take periodic images of the tanker aircraft. The relative navigation system will generate predicted images of the tanker based on the current state estimate and covariance. These predicted images will be compared to the measured image, to provide a virtual measurement as feedback to the system. More details of how the system creates a predicted image will be discussed in Section 2.9

For analyzing image processing techniques, two specific formation positions were examined: the pre-contact and contact positions. At the pre-contact position, the receiver aircraft is approximately 50 feet behind and 10 feet below the tip of the refueling boom. In the contact position, the receiver aircraft is positioned just below the fixed boom, with its refueling receptacle directly in line with the boom extension. This is the optimal refueling location. Figure 2.4 presents all typical formation positions during aerial refueling.

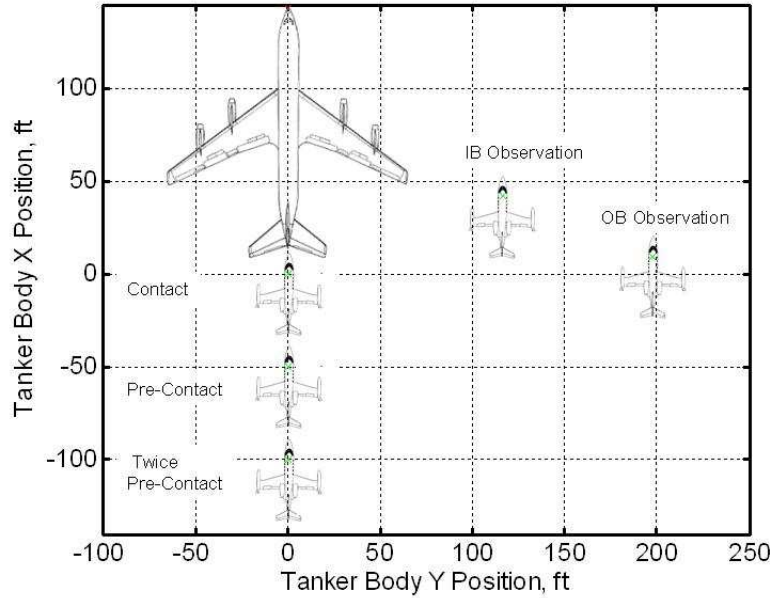


Figure 2.4: Diagram of typical formation positions during aerial refueling. This research will focus on the pre-contact and contact positions [3].

Figure 2.5 shows example infrared images from the receiver camera perspective while in the pre-contact and contact positions. The relative positioning accuracy while the receiver aircraft is in pre-contact, contact, and transitioning between these two points is absolutely critical. Due to the close proximity of the two aircraft during this portion of the refueling operation, any positioning errors could be a safety hazard. Thus, these are optimal positions to evaluate the accuracy of an image-aided algorithm. The simulations presented in Chapter IV will focus on these formation positions.

The key goal for image processing in this research is to determine some type of metric between the predicted and true images. The system must have some way to determine if there are errors between the two images, and how large those errors are. Exactly how the system uses these errors as measurement updates will be discussed in Section 3.2.2. Three image comparison metrics are considered in this research: a sum-squared difference of intensity, a magnitude of gradient comparison, and a gradient with threshold technique. These approaches attempt to quantify the error between a given predicted image and the truth image. Each approach is described in the paragraphs that follow.

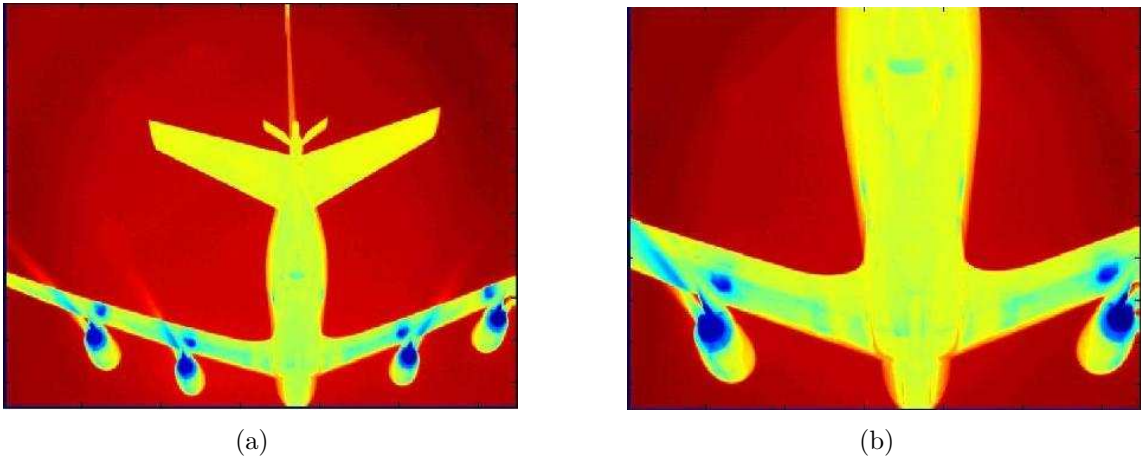


Figure 2.5: (a) Example infrared image of the tanker as seen from the receiver at the pre-contact position.  
(b) Example infrared image of the tanker as seen from the receiver at the contact position



*2.8.1 Sum Squared Difference.* The two images are first converted to grayscale and the respective image intensities are normalized. This normalization is computed by first subtracting the mean image value ( $\bar{\mathbf{I}}$ ) from each pixel, then dividing by the standard deviation ( $\sigma$ ) of the pixel intensities. For a given image,  $\mathbf{I}$ , the normalization calculation is shown in Equation 2.37.

$$\mathbf{I}_{norm} = \frac{\mathbf{I} - \bar{\mathbf{I}}}{\sigma_I} \quad (2.37)$$

In order to avoid aliasing effects, the image is passed through a filter that applies a Gaussian blur to smooth the edges of the aircraft. This Gaussian blur (mean = 0,  $\sigma = 1$  pixel) will be applied during all image-processing techniques.

Now, to find the sum-squared difference,  $c$ , between an image  $\mathbf{I}$  and another image  $\mathbf{N}$  (each with  $n$  pixels), the following expression is calculated:

$$c = \sum_{i=1}^n (\mathbf{I}_i - \mathbf{N}_i)^2 \quad (2.38)$$

Conceptually, the sum-squared difference approach simply compares the differences in intensity between the pixels in one image with the corresponding pixels in another image. This technique would work well when there is a distinct contrast in intensity between the aircraft portion of the image and the background portion of the image. This is indeed true in the case of infrared imaging and the images generated by predictive rendering (Section 2.9).

*2.8.2 Magnitude of Gradient.* A second approach considered for image correlation is to convert the images to gradient space. A gradient is a mathematical calculation to determine the change in image intensity between adjacent pixels. Areas containing a sudden change in pixel intensity indicate edges or corners. In this research, these typically exist at the edges of the tanker aircraft in the image. Applying a special image processing filter in **Matlab**<sup>®</sup> known as the Prewitt filter

(Equation 2.39) to the image, one can determine the horizontal gradient matrix ( $\mathbf{G}_x$ ) for the image.

$$\mathbf{F}_{prewitt} = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} \quad (2.39)$$

Simply taking the transpose of this filter will yield the gradient matrix for the vertical direction ( $\mathbf{G}_y$ ). The overall gradient magnitude matrix in the image is calculated as:

$$\mathbf{G}_{mag} = \mathbf{G}_x^2 + \mathbf{G}_y^2 \quad (2.40)$$

This magnitude is then normalized across the image, similar to the calculation above for sum-squared difference.

$$\mathbf{G}_{norm} = \frac{\mathbf{G}_{mag} - \overline{\mathbf{G}_{mag}}}{\sigma_{\mathbf{G}_{mag}}} \quad (2.41)$$

*2.8.3 Magnitude of Gradient with Threshold.* As a third approach, the same magnitude of gradient is computed for each image, but the magnitudes are then filtered based on a threshold technique. Through experimentation, a threshold is found that will amplify strong areas of gradient magnitude. Each pixel in the image is compared to the threshold value and is set to a value of zero if below the threshold, or set to a value of one if above the threshold. This will remove the few small, unwanted edges (any background clutter or small features on the surface of the aircraft), as well as amplify the desired edges of the aircraft outline. A sample image (after implementing a threshold) is shown in Figure 2.6.



Figure 2.6: Example image after applying magnitude of gradient with threshold. The threshold on the gradient amplifies the edges of the tanker while removing any unwanted gradient due to background clutter or lens reflection.

## 2.9 Predictive Rendering Algorithm

*2.9.1 Motivation.* The basic premise of vision-aided navigation is to extract as much information as possible from digital images. In this research application, the relative distance and attitude between the two aircraft must be determined based on images taken from the wing aircraft imaging sensor. In order to extract this information, there must be some initial prediction of what the image should look like. As discussed in the previous section, an image processing algorithm will compare a predicted image with the truth image in order to provide feedback to the filter. Without a prediction for comparison, the truth image alone does not reveal any specific metric of relative distance and attitude.

The predictive rendering algorithm generates a predicted image using a three-dimensional structural model of the lead aircraft combined with an *a priori* estimate of the relative pose between the imaging sensor and the lead aircraft.

Many image-aided navigation techniques for aerial refueling focus on predicting and tracking specific features (edges or corners) on the tanker aircraft. This can be a computational burden on the system because of the required processing to track and associate multiple features throughout the filter execution. Furthermore, there is the possibility of misassociating a detected feature with an incorrect predicted feature. If the pose or range estimate is significantly wrong, the portion (or window) of the image where the filter predicts a specific feature could potentially contain a different feature of the aircraft. This can lead to even larger errors in the relative navigation solution if the feature detection algorithm associates the wrong detected feature with a predicted feature. In other words, potential errors exist with predicting a certain feature in a small window of the image and finding the wrong feature. Similarly, predicting a feature in a specific portion of the image and detecting no features also leads to errors [15].

In order to simplify this problem, the predictive rendering approach seeks to generate the *entire* image of the tanker and compare it with the entire image taken from a digital camera using image-processing techniques. This eliminates the need to track specific features on the aircraft and also eliminates the possibility of feature misassociations. Additionally, this technique does not require modification of the tracking processes when certain features of the tanker are no longer visible to the camera. For example, when transitioning from the pre-contact to the contact position, the outer extremities of the lead aircraft (i.e., wingtips and tail section) slowly move out of the camera's field of view. A feature detection algorithm would have to discard these features as they moved out of the image and rely on new features (if they exist and are detectable). Images of the tanker taken during these two formation positions are shown in Figure 2.5, presented earlier in this chapter.

Additionally, the predictive rendering technique will give the relative navigation algorithm a modular design. The fundamental algorithm will stay the same regardless of which lead aircraft the wing aircraft is following in the formation. A 3D model of a different tanker aircraft (e.g. KC-10 Extender) could be substituted in for the

KC-135R, and the relative navigation solution would function the same. Furthermore, as long as a 3D model is available, virtually any aircraft could be used as the lead aircraft for other applications of autonomous formation flight.

*2.9.2 Implementation.* In this research, the rendering algorithm was implemented in **Matlab**<sup>®</sup> using the Virtual Reality (VR) Toolbox and a three dimensional model of the KC-135R Stratotanker. The VR toolbox uses OpenGL to render scenes of interest. The VR environment simply needs a translation vector and rotation vector to adjust the location of the 3D model within the window. This approach was used due to the ease with which **Matlab**<sup>®</sup> simulations could be developed that can directly interface with the rendering tool. Though the VR toolbox cannot render at a frame rate as high as more powerful, commercial rendering tools, it has adequate performance to prove the concept of this research. It is not an adequate approach for real-time operation. For example, if the system captured a truth image at a rate of 1 Hz, the rendering tool would be required to generate many predicted images at a very high frame rate. These images would then have to be processed and used for an update to the filter, all in enough time for the system to receive another image update one second later. More on this update process will be discussed in Section 3.2.2.

The relative pose of the lead aircraft is defined as the position and orientation of the lead body frame relative to the camera frame on the wing aircraft. This position and orientation is calculated by:

$$\mathbf{s}^c = \mathbf{C}_W^c [\mathbf{C}_n^W \mathbf{C}_e^n (\mathbf{p}_L^e - \mathbf{p}_W^e) - \mathbf{p}_c^W] \quad (2.42)$$

$$\mathbf{C}_L^c = \mathbf{C}_W^c \mathbf{C}_n^W \mathbf{C}_L^n \quad (2.43)$$

These relative position and orientation parameters are then provided to the predictive rendering algorithm. The VR tool then generates a 3D rendering of the predicted image for any desired translation and rotation. A sample rendering is shown in Figure 2.7.



Figure 2.7: Sample 3D Rendering of Tanker Aircraft. **Matlab**<sup>®</sup> Virtual Reality environment renders a predicted image of the lead tanker from the perspective of the receiver aircraft.

The simulation environment was validated using data collected during flight tests by the Northrop Grumman Corporation. These data sets included time-space positioning information (TSPI) data for both the receiver and tanker aircraft, as well as infrared imaging from a camera mounted inside the nosecone of the receiver. The camera was mounted with a pitch angle of  $+28^\circ$  in order to keep the tanker in its field of view while the receiver aircraft was below the tanker in the contact position. The specific type of imaging sensor used was Long Wave Infrared (LWIR). This type was chosen during these data collection flights due to the ease of locating the tanker aircraft in the infrared image in all lighting and weather conditions [3]. An image of the camera mounted in the nose of the receiver aircraft is shown in Figure 2.8.

A sample predicted image from the simulation is shown in Figure 2.9. Here, the predicted location of the tanker (with an intentional error for easier distinction) is rendered in the same figure as the true infrared image of the tanker. The predicted and measured images must be compared in order to determine the accuracy of the *a priori* state estimate. This comparison metric is generally termed as the residual between predicted and measured values and is described in Section 3.2.2.



Figure 2.8: Orientation of the LWIR Camera when mounted in the nose of the receiver aircraft during data collection flights. The camera has a pitch angle of  $+28^{\circ}$ .

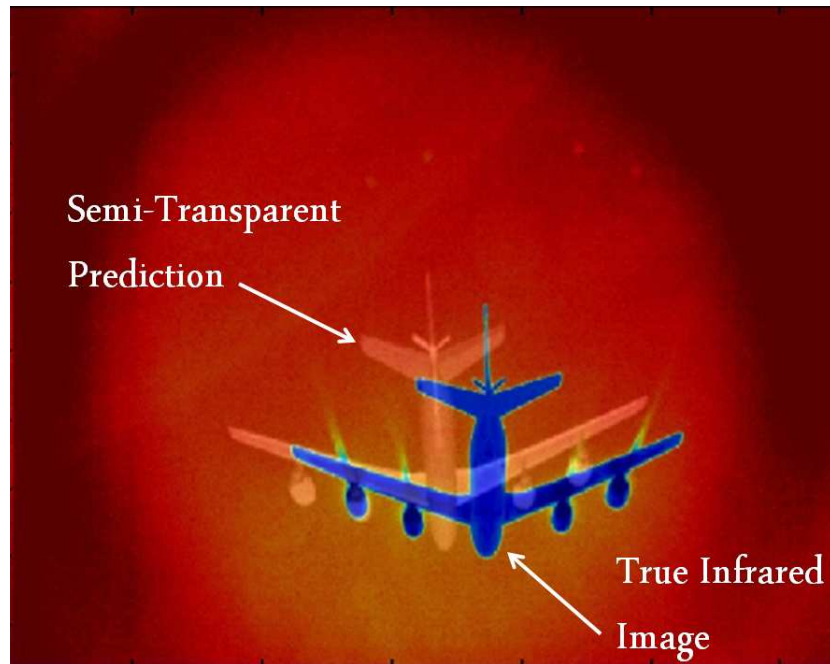


Figure 2.9: Rendered Image Prediction (Semi-Transparent) with True Infrared Image. A semi-transparent rendering of the predicted tanker location is superimposed on an actual infrared image from the receiver camera.

## 2.10 Previous Research Efforts

This section highlights previous work done regarding autonomous aerial refueling, including all significant approaches (GPS, vision-aided, etc.) to date.

*2.10.1 Ross/Spinelli Formation Flight Controller.* The goal of the Ross and Spinelli research was to develop a system that would demonstrate fully-autonomous formation flight and accurate relative position to a lead tanker aircraft. This involved maintaining proper position through all major formations associated with inflight refueling (wing, contact, and pre-contact) [13, 16]. The fundamental approach for positioning (developed by Spinelli) was to use differential GPS (DGPS) between the two aircraft. Using a wireless datalink, the receiver used the tanker position as a local truth reference, and through double-differencing, obtained its position relative to the tanker at centimeter level accuracy [16].

Ross successfully created a flight controller that maintained relative position optimized for  $15^0$  of bank, but capable of up to  $30^0$ . The system under test was a Calspan Learjet as the receiver aircraft with a C-12 aircraft as the surrogate tanker. Figure 2.10 shows the two aircraft in formation during one of the flight tests. This research successfully demonstrated that GPS is one valid approach for accurate relative positioning. Ross also noted that if an optical system could keep the receiver aircraft within a five foot envelope of the nominal refueling position, the human boom operator can correct the error and still successfully make contact with the refueling boom [13].

*2.10.2 Spencer Optical Tracking for Relative Positioning.* Using feature extraction, matching, and validation, Spencer aimed to determine relative position between two aircraft using optical techniques. The fundamental feature extraction technique was the Harris corner detector. For navigation, the system attempts to find and propagate the most likely association feature to the next frame using a





Figure 2.10: Image of Calspan Learjet and C-12 surrogate tanker aircraft during Ross/Spinelli Formation Flight Test

traditional Kalman Filter. The system then tries to match a detected corner or edge from the image with a predicted feature of the target [15].

The system used pose estimates from the inertial measurement unit (IMU) coupled with the measurement update from the Kalman filter's predictions of the feature locations. The radial projection error (between projected features and actual locations) was approximately five pixels for relative distance between 60-300 feet, and approximately 8-12 pixels inside of 60 feet during Spencer's flight tests. Due to this error, a different type of feature extraction or image matching method could potentially allow this approach to more accurately determine the relative position and also be more computationally tractable [15].

*2.10.3 Vision-Based Sensor and Navigation System.* Another visual tracking system for refueling was developed, in concept, at Texas A&M University, called the Vision-Based Sensor and Navigation System, or *VizNav*. The approach here is to use the probe and drogue technique of inflight refueling, with several infra-red light emitting diodes (LEDs) affixed to the drogue chute in a known orientation. When the receiver aircraft approaches the drogue, the system aboard the receiver aircraft

establishes a wireless datalink with the electronics mounted on the drogue. The system then sends commands to the drogue electronics which then activate each LED in a precise, pre-determined sequence. The optical sensor aboard the receiver aircraft can detect and distinguish each LED and thus determine a relative distance and orientation to the drogue [18].

Though a very interesting and seemingly robust approach, this system does demand the ability to establish a reliable datalink between the aircraft and the drogue system. A suitable corollary with boom refueling isn't readily apparent, since with the boom approach the receiver aircraft does not have a visual target (the drogue) to intercept. Furthermore, this approach would require undesirable and costly modifications (installing LEDs) to the tanker fleet [18].

*2.10.4 DARPA/NASA Dryden Probe and Drogue Refueling.* The autonomous aerial refueling system developed under a joint Defense Advanced Research Projects Agency (DARPA) and National Aeronautics and Space Administration (NASA) test at the NASA Dryden test facility involved GPS, vision, and inertial sensors to determine relative position between a refueling drogue from a Boeing 707/300 and an F/A-18 aircraft.

The system primarily calculated the position of the system using the combined relative GPS and INS measurements. Once within visual reference (approximately 80-120 ft), the vision system would track the drogue using an *a priori* model of the drogue shape. The test established an autonomous, hands off, repeatable demonstration of positive contact between the receiver aircraft and refueling receptacle [7].

*2.10.5 Perugia University and West Virginia University.* Perugia University and West Virginia University have many research contributions in the autonomous aerial refueling problem [4, 5, 14]. These efforts focus on comparing different machine vision algorithms for feature detection on the lead tanker aircraft and how to develop robust controllers for autonomously controlling the receiver aircraft in the formation.

These vision-aided approaches all consider tracking specific features on the lead aircraft, sometimes even using infrared optical markers in a known orientation on the tanker. The simulations from Perugia and West Virginia Universities show favorable results, but the research presented in this thesis seeks to avoid tracking specific features of the tanker and/or rely on modifying the tanker fleet by installing optical markers.

### ***2.11 Summary***

This chapter has highlighted key mathematical and navigation background that will be necessary for thorough understanding of Chapter III. The EKF basic discussion will be used to develop relative navigation filters that use INS sensors for propagation and images as virtual measurement updates.

The related research presented in this chapter summarizes key work and successes already accomplished to solve the autonomous aerial refueling problem. The algorithm presented in the next chapter seeks to complement these efforts to obtain a robust, image-aided navigation solution.

### III. Algorithm Development and Implementation

This chapter will describe the algorithm used to develop a vision-aided relative navigation system. First, the dynamics model is developed for both aircraft in the formation. Several image processing techniques are presented to determine the accuracy of the rendering. Finally, the Extended Kalman Filter implementation is discussed, including the algorithm to use images as the measurement update.

#### 3.1 Modeling Aircraft Dynamics

Before constructing a filter to estimate the relative navigation solution, the system dynamics must be accurately modeled. It is assumed that an INS is available on the wing aircraft, so the wing dynamics are modeled based on the nonlinear INS mechanization equations and associated statistical error model.

*3.1.1 Wing Aircraft Dynamics.* The model used in this research builds upon the mechanization development presented in [19]. The accelerometer measurement model is:

$$\mathbf{f}_m^W = \mathbf{f}^W + \mathbf{a}^W + \mathbf{w}_a^W \quad (3.1)$$

Recall from section 2.3 that the  $L$  and  $W$  notation refers to the lead and wing body frames, respectively. The measured specific forces in the wing frame ( $\mathbf{f}_m^W$ ) are the sum of the true specific forces in the wing frame ( $\mathbf{f}^W$ ), the accelerometer biases ( $\mathbf{a}^W$ ), and additive white Gaussian noise ( $\mathbf{w}_a^W$ ). The gyroscope measurement model is:

$$\boldsymbol{\omega}_{ib_m}^W = \boldsymbol{\omega}_{ib}^W + \mathbf{b}^W + \mathbf{w}_b^W \quad (3.2)$$

The measured body rotation rates with respect to inertial space ( $\boldsymbol{\omega}_{ib_m}^W$ ) are the sum of the true body rotation rates ( $\boldsymbol{\omega}_{ib}^W$ ), the gyroscope biases ( $\mathbf{b}^W$ ), and additive white Gaussian noise ( $\mathbf{w}_b^W$ ). The biases for both the accelerometers and the gyroscopes are modeled as first-order Gauss-Markov processes:

$$\dot{\mathbf{a}}^W = -\frac{1}{\tau_a} \mathbf{a}^W + \mathbf{w}_{a_{bias}}^W \quad (3.3)$$

$$\dot{\mathbf{b}}^W = -\frac{1}{\tau_b}\mathbf{b}^W + \mathbf{w}_{bias}^W \quad (3.4)$$

The time constants for the accelerometer and gyroscope are represented by  $\tau_a$  and  $\tau_b$ , respectively.

Fully developed in [19], the error-state model is presented in the following equations. First, the attitude error states are defined in terms of small angles about the north, east, and down directions of the local navigation frame as:

$$\boldsymbol{\psi} = \begin{bmatrix} \psi_n \\ \psi_e \\ \psi_d \end{bmatrix} \quad (3.5)$$

The final attitude error equations (in vector form from [19]) are determined to be:

$$\dot{\boldsymbol{\psi}} = - \left[ (\mathbf{C}_e^n \boldsymbol{\omega}_{ie}^e) \times \right] \boldsymbol{\psi} - \mathbf{C}_W^n \mathbf{b}^W - \mathbf{C}_W^n \mathbf{w}_b^W \quad (3.6)$$

The  $\times$  symbol following a vector indicates the skew symmetric form. The velocity and position error equations (in vector form from [19]) are determined to be:

$$\delta \dot{\mathbf{v}}^n = \mathbf{C}_e^n \mathbf{G} \mathbf{C}_n^e \delta \mathbf{p}^n - 2\mathbf{C}_e^n \boldsymbol{\Omega}_{ie}^e \mathbf{C}_n^e \delta \mathbf{v}^n + (\mathbf{f}^n \times) \boldsymbol{\psi} + \mathbf{C}_b^n \mathbf{a}^b + \mathbf{C}_b^n \mathbf{w}_a^b \quad (3.7)$$

$$\delta \dot{\mathbf{p}}^n = \delta \mathbf{v}^n \quad (3.8)$$

*3.1.2 Lead Aircraft Dynamics.* If assuming that the lead INS measurements are available for the navigation solution, the dynamics are modeled using the same error equations as presented above in the wing dynamics section. Since this research will focus on a completely passive system, the INS measurements are not available and the dynamics are modeled using a stochastic, kinematic motion model. With the knowledge that the lead flight behavior will be relatively benign during a close-proximity refueling maneuver, several of the states can be modeled as first-order Gauss-Markov processes driven by WGN.

The  $x$  velocity dynamics in the lead body frame are modeled as:

$$\dot{v}_x^L = -\frac{1}{\tau_v}v_x^L + w_{v_x}^L \quad (3.9)$$

The variables are tracked in the lead body frame in order to make the first-order Gauss-Markov process a valid modeling tool. The  $y$  and  $z$  velocities in the lead body frame should have very minute variations and each can simply be modeled by a WGN source.

$$\dot{v}_y^L = w_{v_y}^L \quad (3.10)$$

$$\dot{v}_z^L = w_{v_z}^L \quad (3.11)$$

Similar to the  $x$  velocity, the roll ( $\phi_L$ ) and pitch ( $\theta_L$ ) state dynamics are also modeled as first-order Gauss-Markov processes.

$$\dot{\phi}_L = -\frac{1}{\tau_{\phi_L}}\phi_L + w_{\phi_L}^L \quad (3.12)$$

$$\dot{\theta}_L = -\frac{1}{\tau_{\theta_L}}\theta_L + w_{\theta_L}^L \quad (3.13)$$

The time constants for the roll and pitch rates are  $\tau_{\phi_L}$  and  $\tau_{\theta_L}$ , respectively.

The heading ( $\psi_L$ ) differential equation is a function of the amount of roll, the gravity vector ( $g$ ), and the  $x$  velocity as follows:

$$\dot{\psi}_L = \frac{g \tan \phi_L}{v_{x_L}} \quad (3.14)$$

Now to convert to the error state form, mixed partial derivatives of each equation above are taken with respect to each of the state variables (commonly called the Jacobian). The lead velocity error state variable dynamics (in vector form) are

computed as:

$$\delta \dot{\mathbf{v}}_L^L = \begin{bmatrix} -\frac{1}{\tau v_x} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \delta \mathbf{v}_L^L \quad (3.15)$$

The lead position error state variable dynamics (in the navigation frame) are simply a function of the velocity:

$$\delta \dot{\mathbf{p}}_L^n = \mathbf{C}_L^n \delta \mathbf{v}_L^L \quad (3.16)$$

The attitude error state variable dynamics are as follows:

$$\begin{bmatrix} \delta \dot{\phi}_L \\ \delta \dot{\theta}_L \\ \delta \dot{\psi}_L \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ \frac{-g \tan \phi_L}{v_{x_L}^2} & 0 & 0 \end{bmatrix} \delta \mathbf{v}_L^L + \begin{bmatrix} -\frac{1}{\tau_{\phi_L}} & 0 & 0 \\ 0 & -\frac{1}{\tau_{\theta_L}} & 0 \\ \frac{g}{v_{x_L} \cos^2 \phi_L} & 0 & 0 \end{bmatrix} \begin{bmatrix} \delta \phi_L \\ \delta \theta_L \\ \delta \psi_L \end{bmatrix} \quad (3.17)$$

*3.1.3 State-Space Model.* With all of the error state differential equations now written for the entire system, the overall state-space model can be formed. This will be the nonlinear model used in the propagation step of the Kalman filter.

The error state vector,  $\delta \mathbf{x}$ , consists of the position ( $\delta \mathbf{p}_W^n$ ), velocity ( $\delta \mathbf{v}_W^n$ ), attitude ( $\boldsymbol{\psi}_W$ ), accelerometer bias ( $\delta \mathbf{a}^W$ ), and gyroscope bias ( $\delta \mathbf{b}^W$ ) for the wing vehicle, as well as the position ( $\delta \mathbf{p}_L^n$ ), velocity ( $\delta \mathbf{v}_L^L$ ), and attitude ( $\boldsymbol{\psi}_L = [\delta \phi_L, \delta \theta_L, \delta \psi_L]^T$ ) for the lead vehicle. This yields a twenty-four element vector of the form:

$$\delta \mathbf{x} = \begin{bmatrix} \delta \mathbf{p}_W^n \\ \delta \mathbf{v}_W^n \\ \boldsymbol{\psi}_W \\ \delta \mathbf{a}^W \\ \delta \mathbf{b}^W \\ \delta \mathbf{p}_L^n \\ \delta \mathbf{v}_L^L \\ \boldsymbol{\psi}_L \end{bmatrix}_{24 \times 1} \quad (3.18)$$

The noise term consists of the WGN sources that drive the accelerometer measurement ( $\mathbf{w}_a^W$ ), the gyroscope measurement ( $\mathbf{w}_b^W$ ), the accelerometer bias ( $\mathbf{w}_{a_{bias}}^W$ ), and the gyroscope bias ( $\mathbf{w}_{b_{bias}}^W$ ) all for the wing aircraft, as well as the velocity noise ( $\mathbf{w}_{vel}^L$ ), and attitude noise ( $\mathbf{w}_{att}^L = [w_\phi, w_\theta, 0]^T$ ) for the lead aircraft. These elements are combined to form:

$$\mathbf{w} = \begin{bmatrix} \mathbf{w}_a^W \\ \mathbf{w}_b^W \\ \mathbf{w}_{a_{bias}}^W \\ \mathbf{w}_{b_{bias}}^W \\ \mathbf{w}_{vel}^L \\ \mathbf{w}_{att}^L \end{bmatrix}_{18 \times 1} \quad (3.19)$$



The overall, augmented state-space error model can then be written as follows:

$$\begin{aligned}
\delta \dot{\mathbf{x}} = & \begin{bmatrix} \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{C}_e^n \mathbf{G} \mathbf{C}_n^e & -2\mathbf{C}_e^n \boldsymbol{\Omega}_{ie}^e \mathbf{C}_n^e & (\mathbf{f}^n \times) & \mathbf{C}_W^n & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & -(\mathbf{C}_e^n \boldsymbol{\omega}_{ie}^e) \times & \mathbf{0}_3 & -\mathbf{C}_W^n & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & -\frac{1}{\tau_a} \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & -\frac{1}{\tau_b} \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{C}_L^n \mathbf{I}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{F}_1 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{F}_2 & \mathbf{F}_3 \end{bmatrix}_{24 \times 24} \delta \mathbf{x} \\
& + \begin{bmatrix} \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{C}_W^n & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & -\mathbf{C}_W^n & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_1 & \mathbf{0}_3 & \mathbf{G}_1 \end{bmatrix}_{24 \times 18} \mathbf{w}
\end{aligned} \tag{3.20}$$

using the following substitutions:

$$\begin{aligned}
\mathbf{F}_1 &= \begin{bmatrix} -\frac{1}{\tau_{vx}} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} ; \mathbf{F}_2 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ \frac{-g \tan \phi_L}{v_{xL}^2} & 0 & 0 \end{bmatrix} \\
\mathbf{F}_3 &= \begin{bmatrix} -\frac{1}{\tau_{\phi_L}} & 0 & 0 \\ 0 & -\frac{1}{\tau_{\theta_L}} & 0 \\ \frac{g}{v_{xL} \cos^2 \phi_L} & 0 & 0 \end{bmatrix} ; \mathbf{G}_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}
\end{aligned} \tag{3.21}$$

### 3.2 Image-Aided Filter Implementation

This section will highlight the key details of how the filter propagates and updates during operation.

*3.2.1 Extended Kalman Filter Propagation.* As discussed in Section 2.2.2, the Extended Kalman Filter propagates from one measurement update to the next by linearizing the trajectory about the current, nominal state estimate. After a measurement update is taken, the new state estimate is used to linearize the dynamics presented in the previous section. This linear relationship is then propagated to the next time epoch, to yield the *a priori* estimate. The filter is then ready to receive an update based on the image captured from the digital camera.

*3.2.2 Image Update.* Section 2.8 considered three methods that determine an error between the true image collected by the digital camera, and the predicted image generated from the *a priori* state estimate. This error value is simply a numeric number. It quantifies how well the two images match, but has no information regarding which degrees of freedom have significant errors. Using only the error, one can not readily deduce which direction(s) to move the optimal state estimate so that it will yield a prediction that more closely matches the true image.

By perturbing the estimate (in all six degrees of freedom) about the nominal and generating multiple images, the image update can determine in which direction(s) the estimate differs from the truth. Before these perturbations are started, there must be some metric to regulate how large the perturbations in each degree of freedom will be. If the initial estimate is significantly wrong, then small perturbations might not give the algorithm observability into the appropriate directions that will correct the estimate. Also, if the *a priori* estimate is very accurate, then perturbations that are very large will not provide observability into any very small error corrections that are needed.

The state error covariance matrix ( $\mathbf{P}_{xx}$ ) is used to determine how large the perturbation values should be. States that have large uncertainty should obviously have larger perturbations than those with smaller uncertainty. By taking the Cholesky decomposition of this matrix, the perturbation values for each state can be found that include any cross-coupling covariance effects between states. **Matlab**<sup>®</sup> has a built-in function (*chol()*) to easily yield the Cholesky decomposition:

$$chol(\mathbf{P}_{xx})^T \triangleq \sqrt[{}^c]{\mathbf{P}_{xx}} \quad (3.22)$$

Each degree of freedom is then perturbed about the nominal state estimate, using the magnitudes found from the Cholesky decomposition. These magnitudes are applied as both positive and negative values, so that the estimate is perturbed in all possible directions. Mathematically, each perturbation state vector ( $\hat{\mathbf{x}}_i$ ) is a linear function of each row of the Cholesky decomposition plus the nominal, *a priori* state estimate ( $\hat{\mathbf{x}}$ ):

$$\hat{\mathbf{x}}_i = \sqrt[{}^c]{\mathbf{P}_{xx}} \Big|_i + \hat{\mathbf{x}} \quad (3.23)$$

After each individual perturbation generates a new predictive rendering, an image is captured and compared with the nominal, *a priori* image. This yields an error gradient in all directions about the nominal, serving as the observability function (or measurement) of each perturbation:

$$\hat{\mathbf{z}}_i = h[\hat{\mathbf{x}}_i] \quad (3.24)$$

Once every gradient is found (for both positive and negative perturbations), the values are compared to find the direction that produces the gradient value that is the lowest (i.e., most negative). In other words, moving the predicted image in this direction results in the most error reduction of any of the perturbations. The state estimate is adjusted by the amount of the best perturbation, and the process repeats. The estimate is then perturbed about this new nominal value to again find the direction(s)

that yields the greatest reduction in error. The algorithm tracks the translation and rotation adjustments from the *a priori* estimate throughout the process.

As this process cycles, the predicted image should gradually and methodically move so that it closely aligns with the true image from the digital camera. Once the predicted and true images are closely aligned, then the normal perturbations will not yield a direction in which the error will decrease. Moving the prediction in any direction will only increase the error. At this point, the perturbation values are reduced to a fraction (10%) of their original value, and the process above is repeated. This is a method to fine tune the estimate and lock in the predicted image with the true image as closely as possible.

After the fine tuned perturbations reach the point where any movement yields an increase in error, then the prediction should be closely correlated with the true image. This location is indeed a minimum, but it is potentially only a local minimum. It is possible that the global minimum could be at a different position. Using the state error covariance as the scale with which to perturb helps to mitigate this possibility. If the covariance values are high, then the perturbations should be high enough to find the global minimum. If the covariance is low, indicating the nominal estimate is near the truth, then small perturbations should find the global minimum. The cumulative translation and rotation adjustments from the *a priori* estimate to the closest match yield the measured location and orientation of the lead aircraft.

As shown in Equations 2.13 and 2.18 (for both the traditional and Extended Kalman Filters) a measurement uncertainty matrix,  $\mathbf{R}$ , must be included as part of the measurement update step. In order to find the uncertainty of the measurements taken from the image perturbations, the shape of the error curves must be known. Every error value is stored as the algorithm perturbs the estimate about the nominal until finally locking on to the best matching image. These errors are recorded for each degree of freedom, and sorted based on their relative perturbation value from the nominal. To explain this concept graphically, consider Figure 3.1.

For each degree of freedom, the minimum error should occur where the predicted and true images are precisely aligned. Intuitively, any deviation from this point should cause the error value to increase. The data should approximately align with a parabolic function, with the errors increasing the further one perturbs from the optimal value. This parabolic shape is approximated as a Gaussian distribution that is negated. The variance ( $\sigma^2$ ) of this Gaussian distribution is the measurement uncertainty for that particular degree of freedom. The  $\mathbf{R}$  matrix is formed by placing all six variances along its diagonal. The cross terms of the  $\mathbf{R}$  matrix are assumed to be zero since there is no direct correlation between uncertainties.

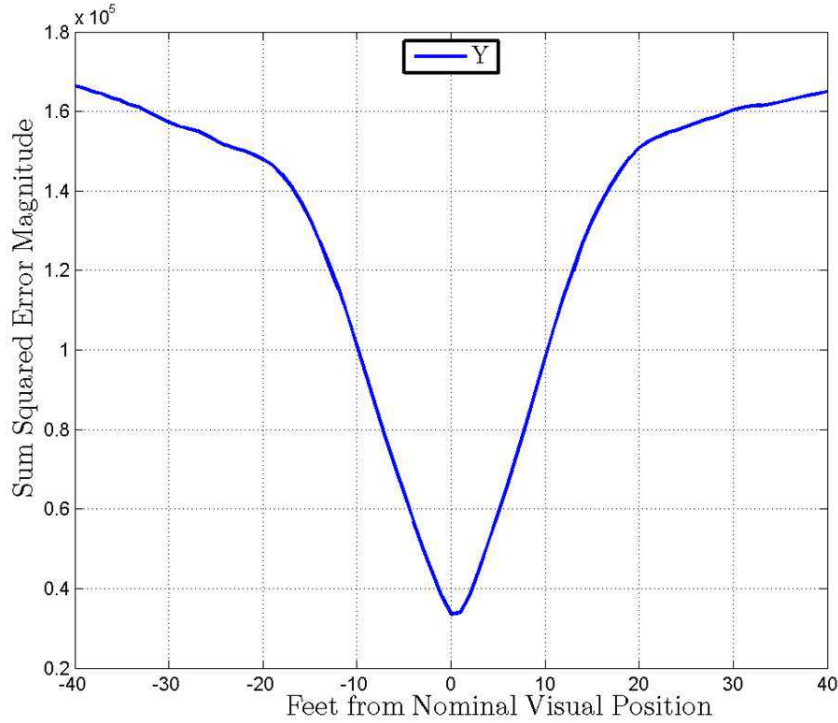


Figure 3.1: Example error distribution about the best image match. The blue curve represents the typical behavior of a single degree of freedom when perturbed about the location where the true and predicted images best match. A negated Gaussian distribution is approximated based on the shape of this curve, whose variance is used as the measurement uncertainty.

### ***3.3 Summary***

This chapter has shown how the relative navigation solution is calculated using an Extended Kalman Filter. The filter propagates the system dynamics over time using INS mechanization for the wing aircraft and first order Gauss-Markov approximations for the lead aircraft. Using a predictive rendering approach, an estimate of the lead aircraft image as seen from the wing onboard camera is generated at the time epochs where a measurement is taken. Through image processing and perturbation techniques, the image update function determines the relative position and translation error between the predicted and true images. This serves as a measurement that is fed back to the EKF, which updates the state estimate. The filter is then ready to propagate forward again and iterate the process.

Based on this development, Chapter IV will present analysis of the various image processing techniques discussed in this chapter, as well as multiple simulations to test the relative navigation solution under various conditions.

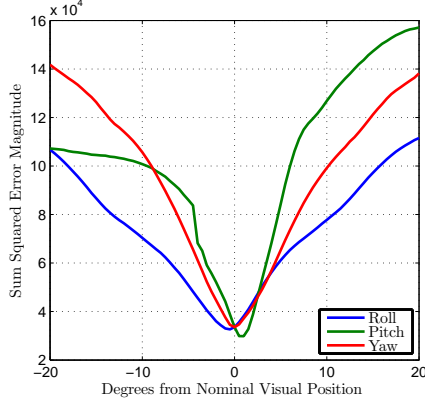
## IV. Algorithm Simulation and Analysis

This chapter has two major focuses. The first portion analyzes the three image processing techniques discussed earlier through numerous image processing experiments. The second portion presents a performance analysis of the Extended Kalman Filter (EKF) developed in Chapter III using simulated trajectories and images.

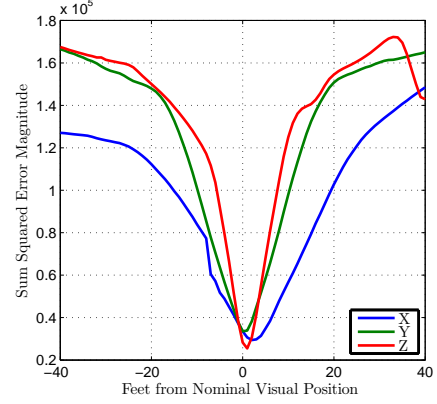
### 4.1 *Analysis of Image Processing Techniques*

This section presents an analysis of the three image processing techniques (sum-squared difference, magnitude of gradient, and magnitude of gradient with threshold) discussed in Chapter III. Simulations were run to determine the observability of translation and rotation errors between two given images using each comparison technique. The results of these simulations will provide insight into how well each technique will detect errors between the predicted and true images. All three techniques are analyzed at the pre-contact and contact positions of the refueling formation, using infrared images from the Northrop Grumman test flights as truth images. The plots shown in this section can be conceptually thought of as cost functions. The global minimum should be the optimal prediction (i.e., the nominal estimate where the predicted image exactly matches the truth image).

*4.1.1 Sum-Squared Difference.* Considering each degree of freedom for a specific time epoch, the roll, pitch, and yaw are perturbed from  $-20^0$  to  $20^0$  from the nominal, and the  $x$ ,  $y$ , and  $z$  body axes are perturbed from  $-40$  feet to  $40$  feet from the nominal position. These plots are shown in Figure 4.1. The values in the plots are the sum-squared difference (or error) between a predicted image and the truth image. One can see that the sum-squared difference approach shows a smooth rise in the error as the perturbations deviate away from the nominal location (0 on each plot's horizontal axis). In certain degrees of freedom, the minimum error does not precisely occur at zero. This is due to small discrepancies in the rendered model dimensions compared to the images of the true tanker. For example, differences in



(a)



(b)

Figure 4.1: (a) The predicted image is independently perturbed in roll, pitch, and yaw about the nominal pre-contact position. The minimum error (using sum-squared difference) should occur where the images are best correlated. (b) The predicted image is independently perturbed in the  $x$ ,  $y$ , and  $z$  body axes about the nominal pre-contact position. The minimum error (using sum-squared difference) should occur where the images are best correlated.

apparent engine size between the rendered and true images can be observed at some attitudes. A sample surface plot is shown in Figure 4.2. This particular plot shows the relationship between both roll perturbation and body-frame longitudinal ( $x$  axis) perturbation about the nominal estimate. Deviations from the nominal are observable as the surface smoothly slopes to higher magnitudes of error the further the predicted image moves from the nominal image.

The next mode presented shows the correlation between perturbations about the  $x$  axis and perturbations about pitch (Figure 4.3). There is a definite observability issue in this case, since the very large valley would make it difficult for the image processing system to isolate the minimum error. One final mode at the pre-contact position that shows observability issues is the  $z$  axis translation coupled with pitch, which is shown in Figure 4.4. Again, the undulating valley through this surface will cause difficulties in determining the true minimum.



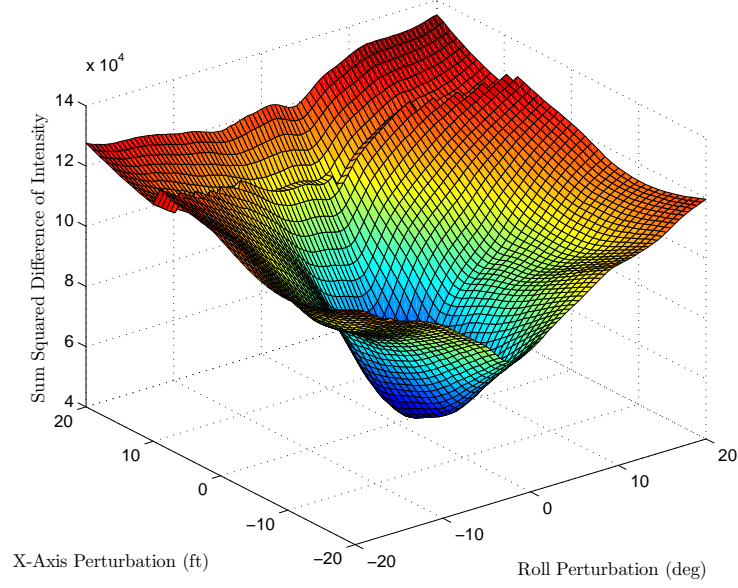


Figure 4.2: Example surface generated using sum-squared difference approach by perturbing both the translation on the  $x$  axis and the amount of roll simultaneously at pre-contact position.

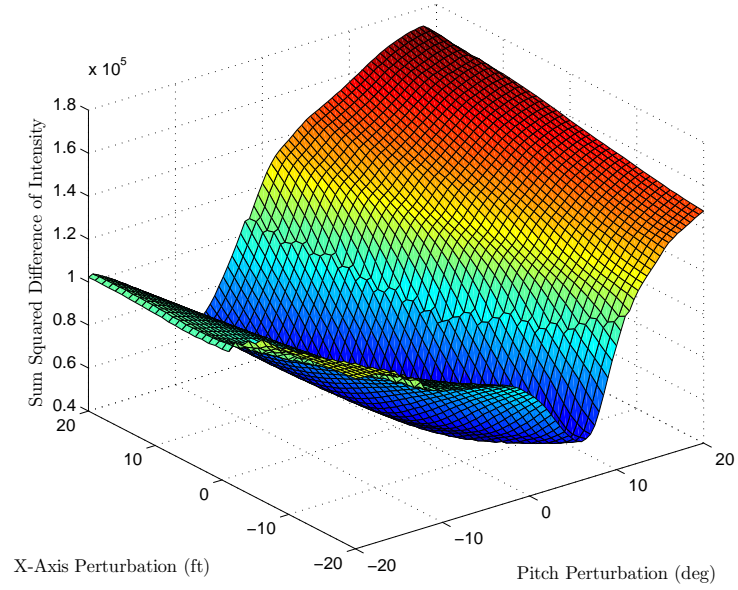


Figure 4.3: Poor observability example surface generated with sum-squared difference approach by perturbing both the  $x$  axis translation and amount of pitch simultaneously at pre-contact position. The large valley makes observability of the optimal, minimum location extremely difficult.

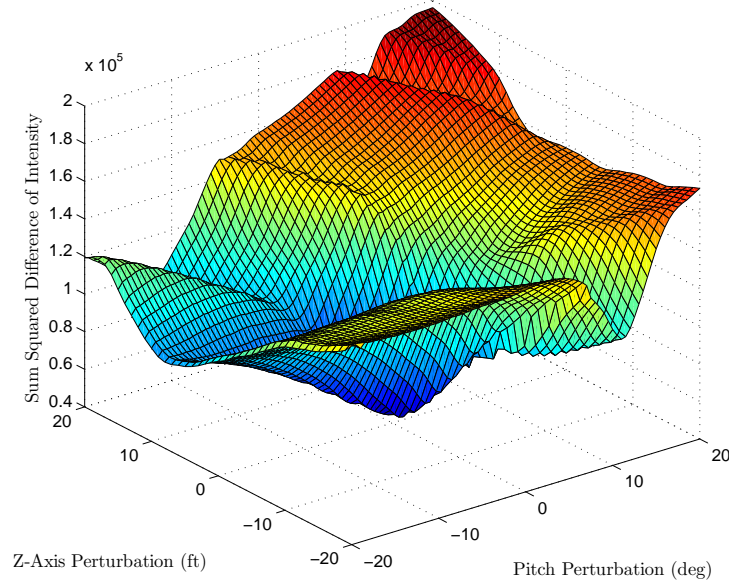


Figure 4.4: Example surface generated with sum-squared difference approach by perturbing the  $z$  axis translation and the amount of pitch simultaneously at pre-contact position. The valley that also exists here makes the minimum difficult to locate.

The additional modes for the other coupled degrees of freedom demonstrate favorable behavior similar to the mode of Figure 4.2, showing good overall observability of errors for this technique.

Now analyzing the contact position for the sum-squared difference approach, two surfaces are presented showing modes of interest. The surface in Figure 4.5 shows the same perturbation as Figure 4.2, with similar, smooth behavior. Figure 4.6 shows the same perturbation as Figure 4.3, again showing poor observability since the majority of the surface is very flat with no distinct minimum.

With this analysis of the sum-squared difference technique as a baseline, the next section will consider the magnitude of gradient approach and analyze its potential error detection capability.

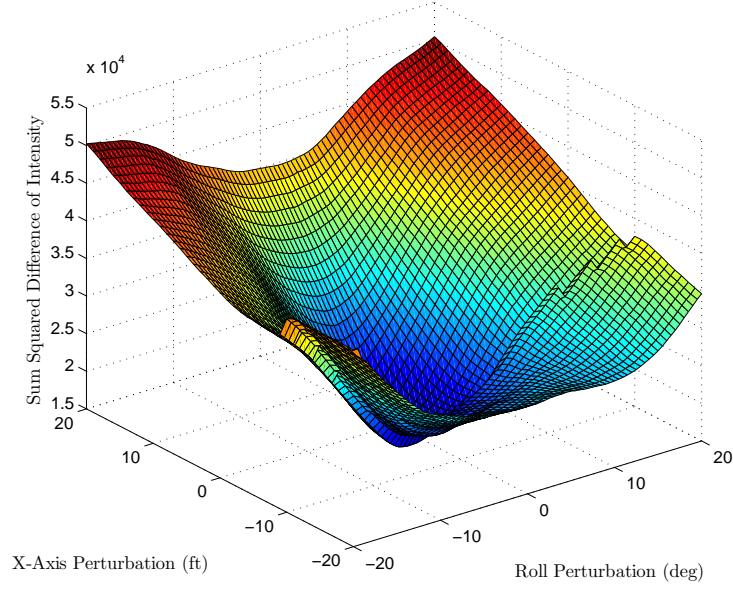


Figure 4.5: Example surface generated using sum-squared difference approach by perturbing both the translation on the  $x$  axis and the amount of roll simultaneously at the contact position.

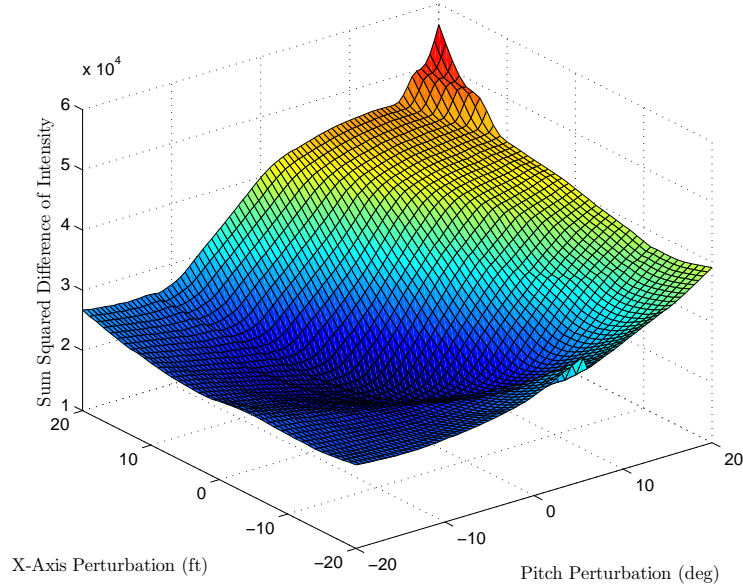


Figure 4.6: Example surface generated using sum-squared difference approach by perturbing both the translation on the  $x$  axis and the amount of pitch simultaneously at the contact position.

*4.1.2 Magnitude of Gradient.* This section will analyze the magnitude of gradient comparison technique. Figure 4.7 shows similar plots as Figure 4.1. Here, one notices that the error increases more rapidly than the sum-squared difference technique as the perturbations vary from the nominal position. However, as one perturbs farther and farther from the nominal, the error begins to oscillate rather than smoothly increase.

Figure 4.8 shows a similar surface as seen in Figure 4.2, but compares magnitude of gradient between the two images. Here, one sees that the error increases more rapidly than the sum-squared difference method when the image deviates from the nominal position. There is a distinct, minimum value indicating the nominal location.

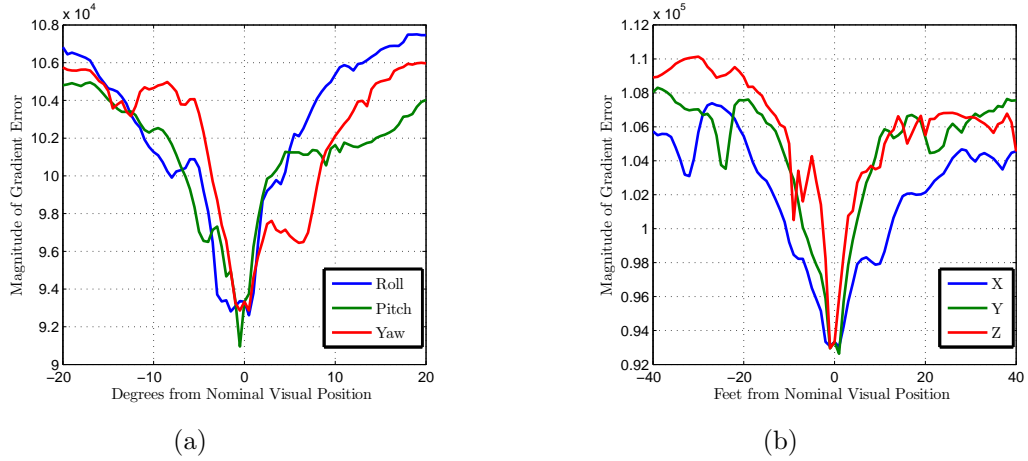


Figure 4.7: (a) The predicted image is independently perturbed in roll, pitch, and yaw about the nominal pre-contact position. The minimum error (using magnitude of gradient) should occur where the images are best correlated. (b) The predicted image is independently perturbed in the  $x$ ,  $y$ , and  $z$  body axes about the nominal pre-contact position. The minimum error (using magnitude of gradient) should occur where the images are best correlated.

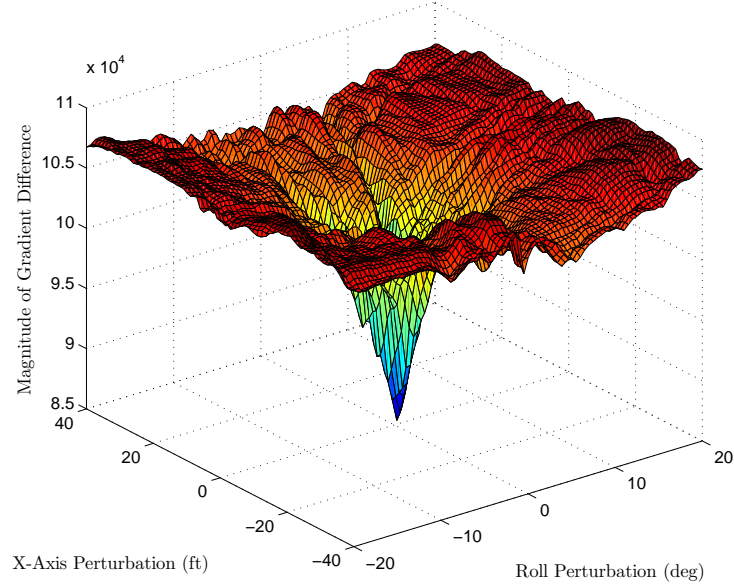


Figure 4.8: Example surface generated using magnitude of gradient approach by perturbing both the  $x$  axis translation and amount of roll simultaneously at the pre-contact position. The sharp minimum indicates good observability.

Additional surfaces are shown in Figures 4.9-4.11. Each of these modes shows numerous local minima, thus making the optimal position difficult to determine. With multiple modes showing this type of behavior, the magnitude of gradient approach alone would not be a robust enough technique for precise error detection.

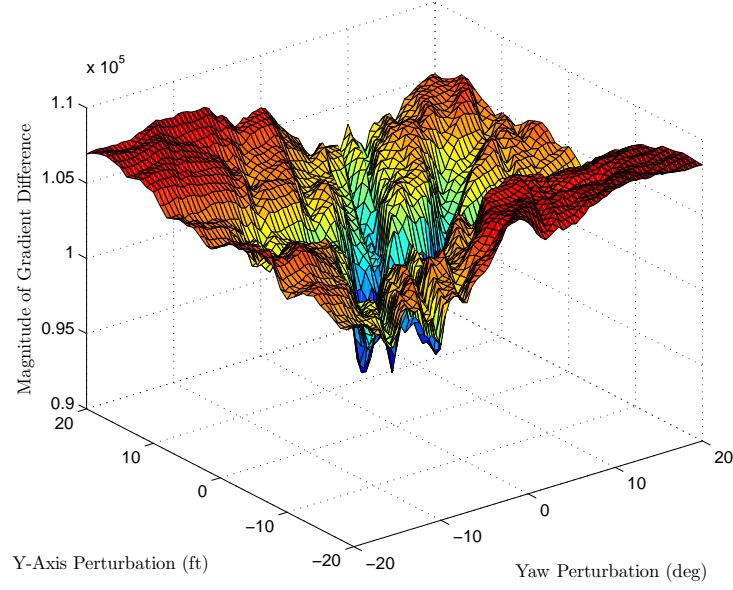


Figure 4.9: Example surface generated using magnitude of gradient approach by perturbing both the  $y$  axis translation and the amount of yaw simultaneously at the pre-contact position. There are many local minima, indicating poor observability.

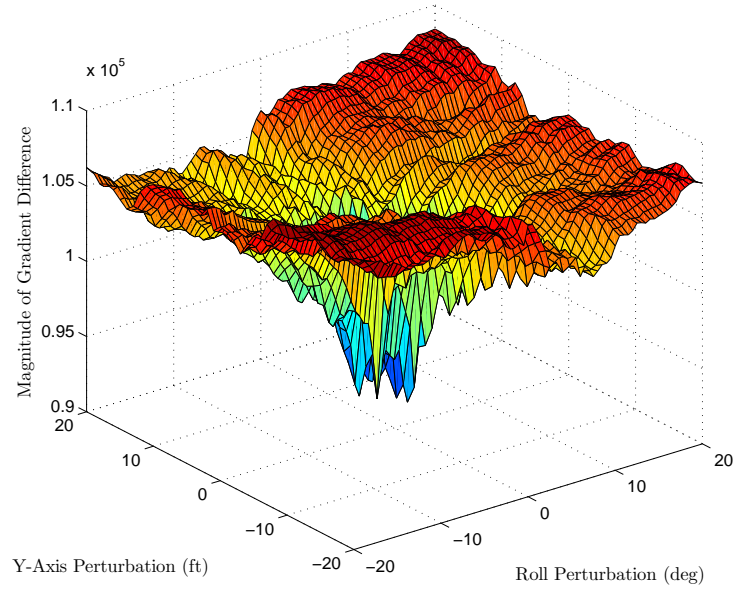


Figure 4.10: Example surface generated using magnitude of gradient by perturbing both  $y$  axis translation and the amount of roll simultaneously at the pre-contact position. Again, multiple minima cause observability problems.

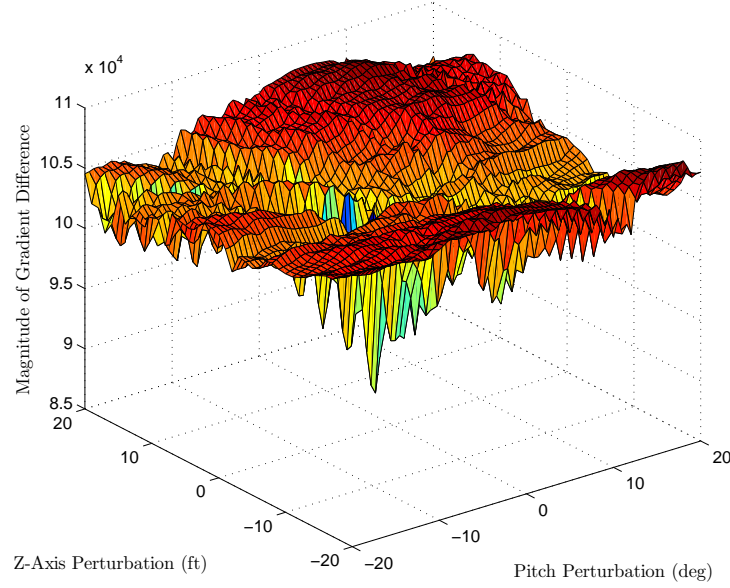


Figure 4.11: Example surface generated using magnitude of gradient approach by perturbing both  $z$  axis translation and the amount of pitch simultaneously at the pre-contact position. Oscillations throughout this mode cause minima across the surface which cause difficulties in image correlation.

Now examining the magnitude of gradient technique while at the contact position, Figure 4.12 shows rather poor observability to the errors, but there is a distinct minimum at the nominal location. If the errors begin very near the optimal position, this approach could find the correct minimum. Figure 4.13 mimics the behavior of Figures 4.3 and 4.6, demonstrating that  $x$  axis translation error coupled with pitch error proves to be difficult to detect.



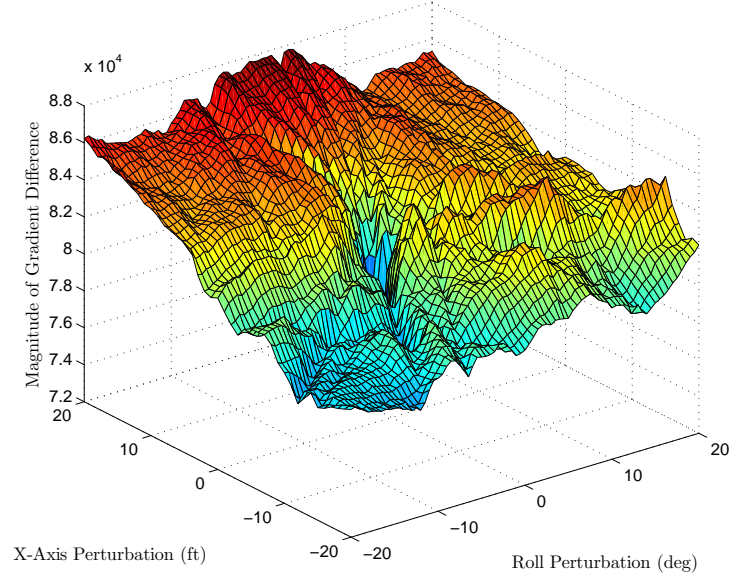


Figure 4.12: Example surface generated using magnitude of gradient by perturbing both  $x$  axis translation and the amount of roll simultaneously at the contact position. There is poor observability across most of the surface.

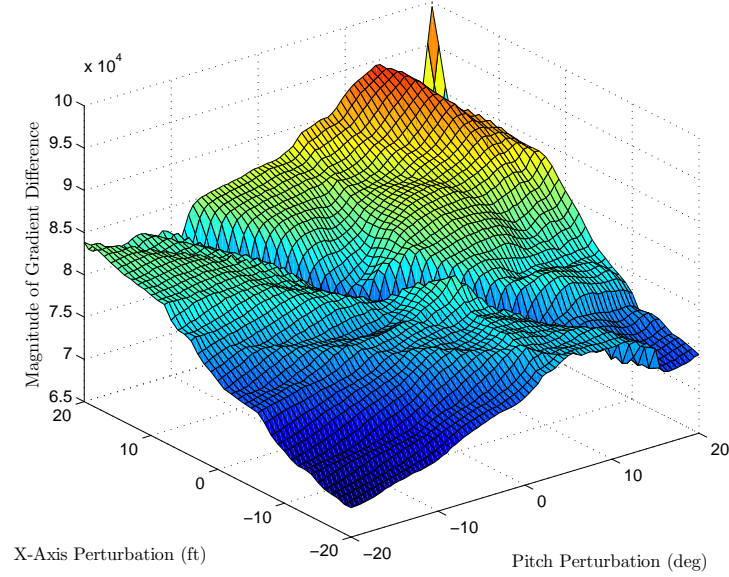


Figure 4.13: Example surface generated using magnitude of gradient approach by perturbing both  $x$  axis perturbation and the amount of pitch simultaneously at the contact position. Similar to Figures 4.3 and 4.6, there is poor observability with this mode, including a valley in the surface.



*4.1.3 Magnitude of Gradient with Threshold.* This section will analyze the magnitude of gradient with threshold comparison technique. Figure 4.14 shows the results when a sample image was perturbed and compared to the truth image using the gradient with threshold technique. One can see that the plots do not have oscillations that are quite as severe as the standard gradient plots shown in Figure 4.7. However, the oscillations are still present, causing local minima at values significantly different from the nominal position. Furthermore, in certain degrees of freedom (such as roll, pitch, and  $x$  axis perturbation), the minimum value is actually skewed by a few degrees/feet.

An example surface generated from the magnitude of gradient with threshold approach is shown in Figure 4.15. This particular surface is not quite as smooth as the sum-squared difference plot of the same mode, but shows a sharper rise away from the nominal position as perturbations are introduced. This surface also does not possess a distinct minimum at the nominal image location, again causing a potential issue

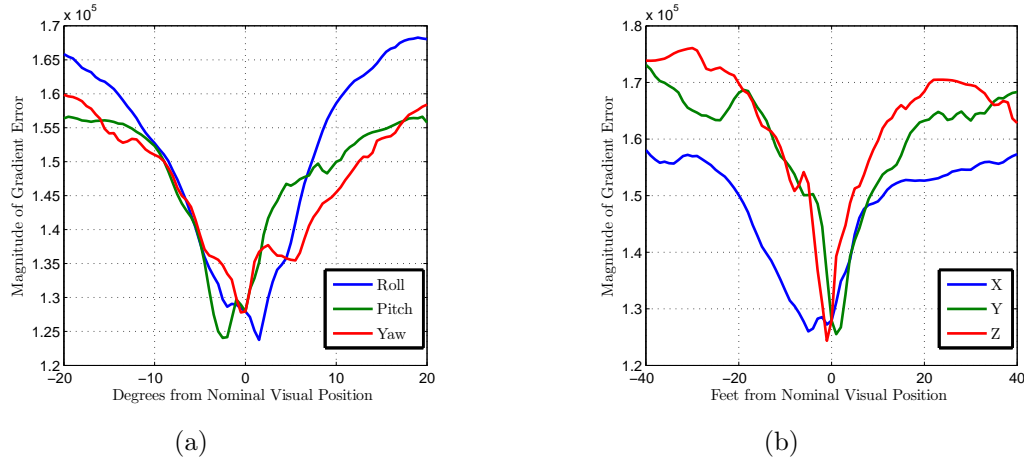


Figure 4.14: (a) The predicted image is independently perturbed in roll, pitch, and yaw about the nominal pre-contact position. The minimum error (using magnitude of gradient with threshold) should occur where the images are best correlated. (b) The predicted image is independently perturbed in the  $x$ ,  $y$ , and  $z$  body axes about the nominal pre-contact position. The minimum error (using magnitude of gradient with threshold) should occur where the images are best correlated.

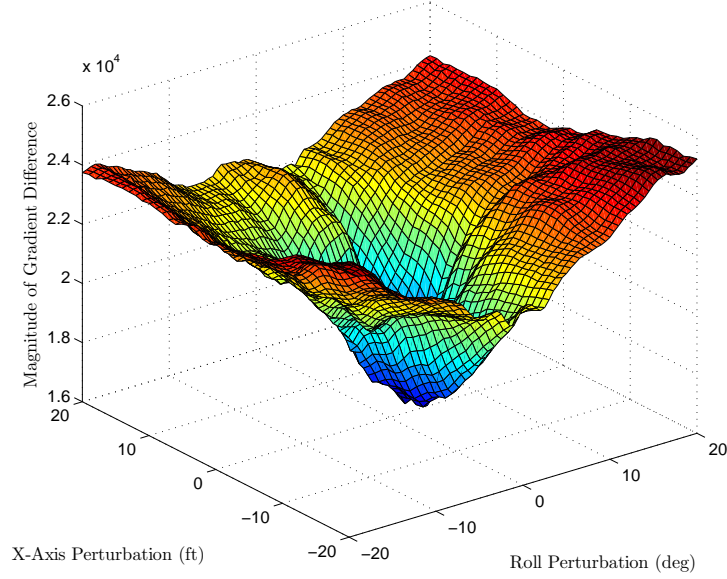


Figure 4.15: Example surface generated using magnitude of gradient with threshold approach by perturbing both the  $x$  axis translation and the amount of roll simultaneously at the pre-contact position. The multiple minima near (0,0) would make the optimal location difficult to detect.

with observability. Furthermore, examining the mode of  $x$  axis perturbation with simultaneous change in pitch, similar results appear as those from the sum-squared difference technique. This surface is shown in Figure 4.16.

Finally, examining the magnitude of gradient with threshold while at the contact position, the first mode presented for comparison is the  $x$  axis translation error coupled with roll error. This surface is shown in Figure 4.17. Very similar to Figure 4.8, there is a very distinct minimum that would easily be detected when near the optimal image location. However, when the errors are larger, the undulations in the outer portions of the surface would make error detection difficult. Furthermore, in Figure 4.18, the  $x$  axis error coupled with pitch error is once again difficult to detect. Similar to previous plots, the undulations and large valley could make the distinct minimum difficult to detect.

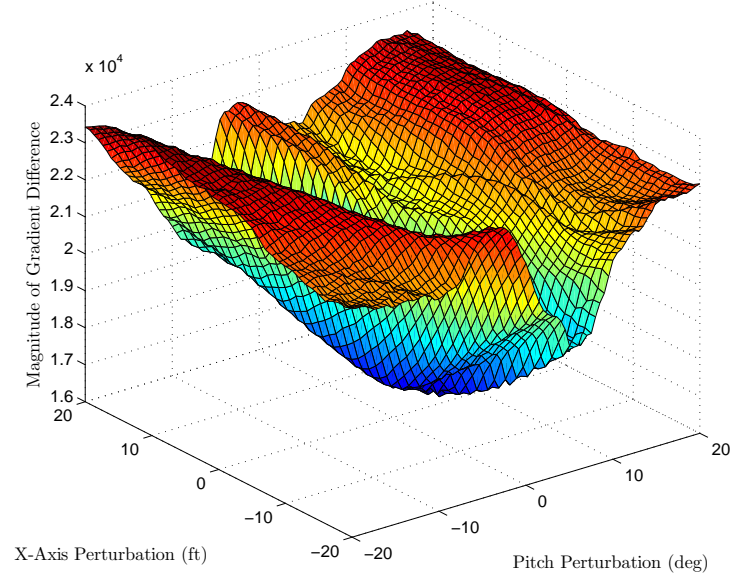


Figure 4.16: Example surface generated using magnitude of gradient with threshold by perturbing the  $x$  axis translation and amount of pitch simultaneously at the pre-contact position. The valley that exists will hinder error detection.

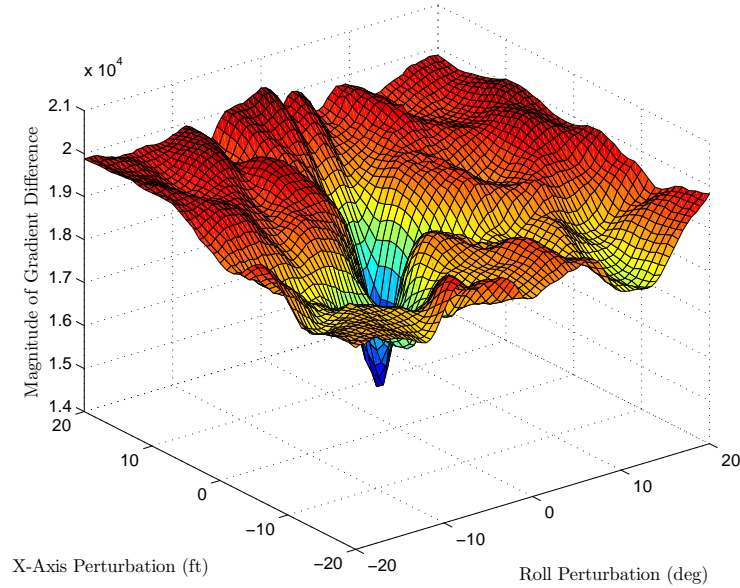


Figure 4.17: Example surface generated using magnitude of gradient with threshold by perturbing the  $x$  axis translation and amount of roll simultaneously at the contact position. There is a distinct minimum when near the optimal location, but large errors would fall on the oscillating outer portion of the surface.

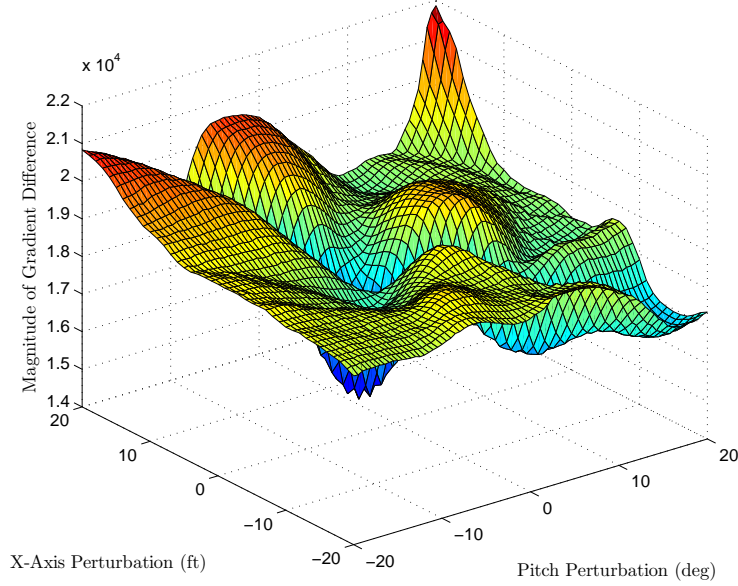


Figure 4.18: Example surface generated using magnitude of gradient with threshold by perturbing both the  $x$  axis translation and amount of pitch simultaneously at the contact position. Observability issues exist due to the oscillations and the valley.

*4.1.4 Analysis.* Although the magnitude of gradient technique generates a distinct minimum in the surface of Figure 4.8, its observability is actually poor in many of the other modes shown in Figures 4.9-4.11. Furthermore, if the predicted image shows large errors (i.e., one or more translation or rotation degrees of freedom are significantly different from the values in the true image), the magnitude of gradient could struggle to identify the actual error. A short example will help illustrate this point.

Considering the two surfaces presented in Figures 4.2 and 4.8, assume that the system predicts a translation in the  $x$  axis that has an error of 20 feet. In the case of the sum-squared difference correlation, if one perturbs the predicted value about this point (i.e., considers points on the surface in the neighborhood of this prediction) and compares errors between the new images and the truth image, a smaller error will definitely be detected and the system can adjust its prediction so that it more closely matches the true image. This process can be repeated until the prediction eventually

moves in the neighborhood of close correlation (near the point (0,0) on the graph) with the true image (as explained in the image update discussion of Section 3.2.2). An observability issue could exist very near the nominal point, however, since this region of the surface is rather flat and the error values vary quite slowly.

With the same conditions in the magnitude of gradient approach, small perturbations about this point on the surface will likely yield very similar results to the predicted point. In other words, all of the points in the neighborhood of this predicted point will yield similar errors. Thus, the system will have difficulty not only detecting that the prediction is incorrect, but also determining which degree(s) of freedom to correct. Considering the magnitude of gradient with threshold approach, the surface shows similar behavior to the sum-squared difference surface, although the errors appear to increase more rapidly than in Figure 4.2. Although this attribute would help with error detection, there is still ambiguity around the nominal point because this portion of the surface contains multiple minima.

Based on the above reasoning, the most robust approach appears to be a combination of the sum-squared differencing technique for coarse error detection, then transitioning to the magnitude of gradient approach for fine tuning. The sum-squared difference technique shows the fewest issues with detecting errors across the entire surface. There are only two modes ( $x$  axis translation error coupled with pitch error and  $z$  axis translation error coupled with pitch error) that show significant error detection issues. This is true for the receiver aircraft in both the contact and pre-contact formation positions. Based on the surfaces generated in the three image processing techniques, there is an evident observability issue with  $x$  axis and pitch.

Upon actually implementing the image update in simulations (to be discussed in detail in the next section), the benefits of using this two-fold approach are negligible in some cases. Through experimentation, the standalone sum squared difference approach performs almost identically to the approach where sum-squared difference is used for coarse acquisition with a transition to magnitude of gradient for fine ad-

justment. However, there are still potential observability issues with the mode of  $x$  body-axis error coupled with pitch error, and using the magnitude of gradient during fine acquisition did present an advantage. The sum-squared difference standalone approach demonstrated some observability challenges in the simulations, which the magnitude of gradient helped to mitigate to some extent. Specific examples of this trend will be discussed in the next section.

## ***4.2 Filter Analysis Using Generated Data***

In order to validate the filter design and image update algorithm, several different scenarios are presented in this section. For each scenario, truth trajectories are generated for both aircraft. For each wing trajectory, simulated inertial measurements are generated for use in the inertial propagation portion of the filter. These inertial measurements are corrupted with noise levels consistent with the grade of the simulated inertial sensor. There are two grades of IMU modeled in the following simulations: a navigation grade Honeywell H764-G, and a tactical grade Honeywell HG-1700. The respective IMU specifications are shown in Table 4.1.

Additionally, based on the relative position and orientation between the two aircraft along the trajectory, renderings are made of the tanker as it should appear from a digital camera on the receiver aircraft. Recall that Figure 2.5 shows the typical image taken by the wing aircraft at the pre-contact position, and Figure 2.7 shows a corresponding rendering at the pre-contact position. To provide the image update function with truth images, renderings are captured to simulate images from the wing onboard camera. Each image is also corrupted with a small amount of translation and orientation noise in order to cause very slight perturbations. Even in smooth air, all aircraft in flight have small movements, so these small perturbations create more realistic images. The specific details, results, and analysis for each scenario are presented in the sections that follow.

Table 4.1: IMU Performance Parameters

| Parameter (Units)                                | HG-1700 | H764-G  |
|--|---------|---------|
| Sampling interval (ms)                           | 10.0    | 3.906   |
| Gyro bias sigma ( $\frac{deg}{hr}$ )             | 1.0     | 0.0015  |
| Gyro bias time constant (hr)                     | 1       | 1       |
| Angular random walk ( $\frac{deg}{\sqrt{hr}}$ )  | 0.3     | 0.002   |
| Gyro scalefactor sigma (PPM)                     | 150     | 5       |
| Accel bias sigma ( $\frac{m}{s^2}$ )             | 0.0098  | 2.45e-4 |
| Accel bias time constant (hr)                    | 1       | 1       |
| Velocity random walk ( $\frac{m/s}{\sqrt{hr}}$ ) | 0.57    | 0.0143  |
| Accel scalefactor sigma (PPM)                    | 300     | 100     |

*4.2.1 Simple Trajectory at Pre-contact Position.* The first simulation considered the receiver aircraft at the pre-contact position, with both aircraft aligned in straight-and-level flight. The filter is initialized with the true initial conditions. The initial state error covariance values for this simulation are given in Table 4.2. The position, velocity, and attitude initial uncertainties are the same for both aircraft.

The EKF is executed using the generated INS measurements for propagation of the wing aircraft navigation state. At a 1 Hz rate, the *a priori* relative navigation state is used to render the estimated position of the tanker, which is then compared to the current truth image. Using the image update function described in Chapter III, the *a posteriori* estimate is computed, and the filter propagates to the next image update (1 second later). In each of the following scenarios, a certain formation position and IMU grade is considered and simulated. A 30 iteration Monte Carlo analysis is presented for each.

*4.2.1.1 Navigation Grade IMU.* As a baseline, the relative errors and covariances are shown for the simulation of free inertial in Figure 4.19. In other words, this represents the typical errors when only relying on the inertial sensors and no measurement updates. The majority of the sample functions are closely aligned with the mean.

Table 4.2: Initial State Uncertainties for Simulations

|  |                         |
|--|-------------------------|
| Position Uncertainty (North, East, and Down)       | $5^2 (m)^2$             |
| Velocity Uncertainty (North, East, and Down)       | $0.1^2 (\frac{m}{s})^2$ |
| Roll ( $\phi$ ) and Pitch ( $\theta$ ) Uncertainty | $1^2 (deg)^2$           |
| Heading ( $\psi$ ) Uncertainty                     | $5^2 (deg)^2$           |

Now using the image update function, Figure 4.20 shows the error in the tanker position with respect to the wing body frame and the standard deviation ( $\pm 1\sigma$ ) for the ensemble. Therefore, this figure shows the difference between the estimated and true position of the lead aircraft with respect to the wing body frame. Figure 4.21 displays the position (estimated minus true) measurement residuals during the simulations. Figure 4.22 displays the attitude (estimated minus true) measurement residuals during the simulations.

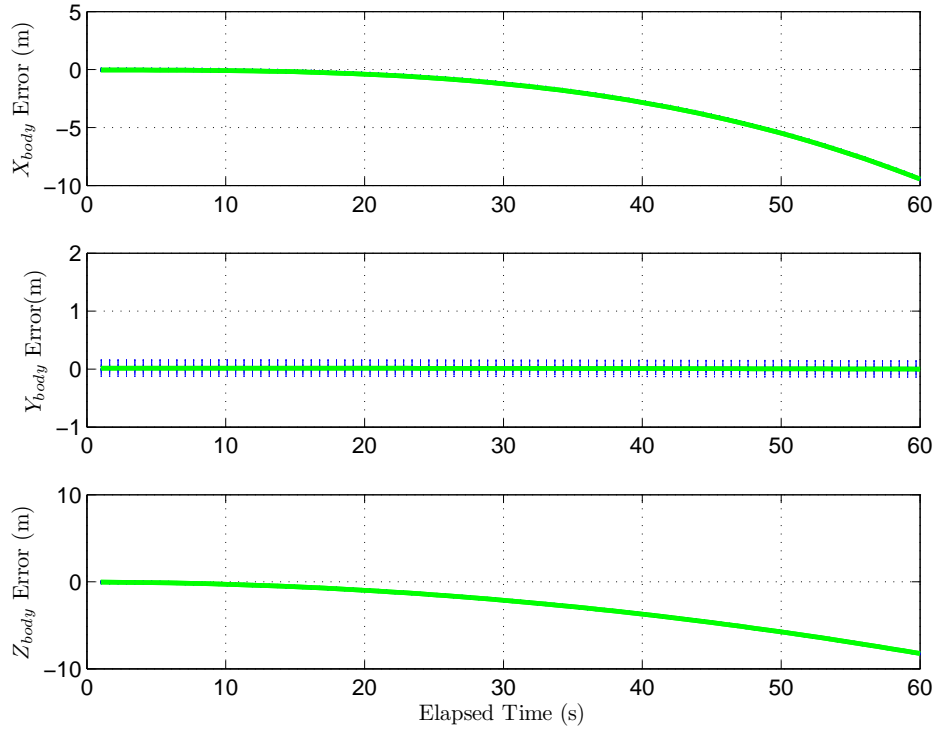


Figure 4.19: The mean relative position error (indicated by the green line) for the Monte Carlo simulation using navigation grade IMU and no image updates, expressed in the wing body frame while at the pre-contact position. The sample functions for the 30 iterations are indicated by the blue, dotted lines.



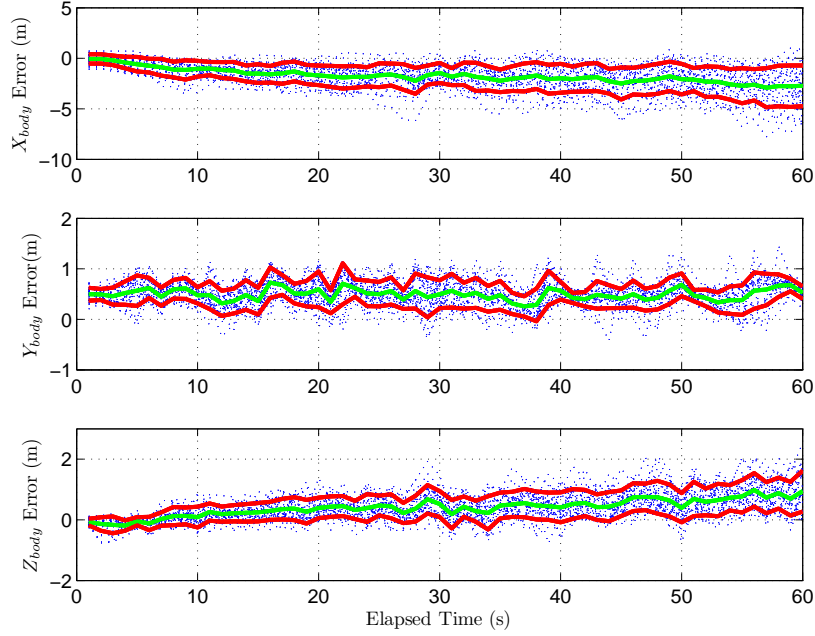


Figure 4.20: Relative position error for the Monte Carlo simulation using navigation grade IMU, expressed in the wing body frame while at the pre-contact position. The relative error sample functions are indicated by the blue, dotted lines. The solid green line is the ensemble mean, and the red lines indicate the ensemble standard deviation (mean  $\pm 1\sigma$  bounds).

The plots indicate the filter can successfully track the location of the tanker for this simple trajectory. The mean  $y$  and  $z$  errors in the wing body frame remain under one meter for the duration of the simulation, while the mean  $x$  errors remain under approximately two meters. The measurement residuals relative to the navigation frame show similar results. Though the sample functions of Figure 4.21 show more variance in the East and Down directions, the mean errors are always less than one meter. This is consistent with the body frame errors presented in Figure 4.20. The attitude residuals show favorable results, with the majority of mean residual values remaining less than one degree. As demonstrated in the previous section, the image update has some observability issues in the  $x$  axis. This is evident in this simulation since the  $x$  axis errors are larger than the other axes.

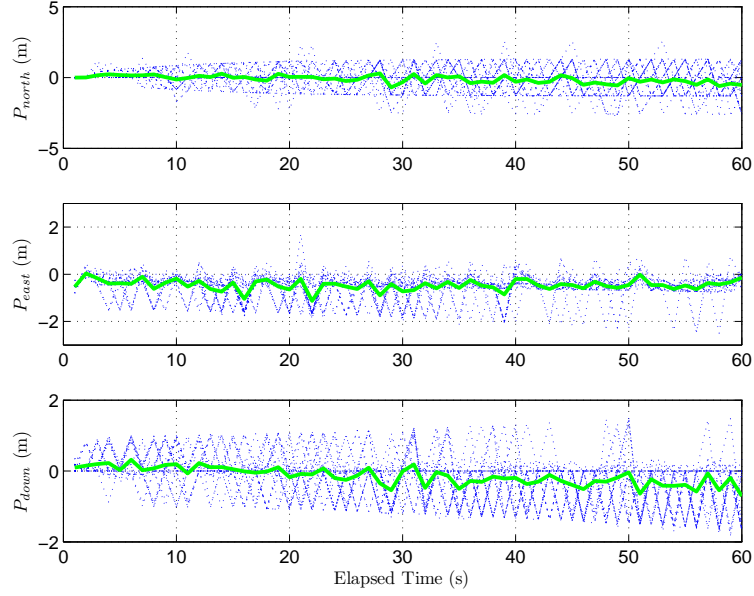


Figure 4.21: Position measurement residuals for the Monte Carlo simulation using navigation grade IMU at the pre-contact position. The sample functions are indicated by the blue, dotted lines. The ensemble mean is indicated by the green line.

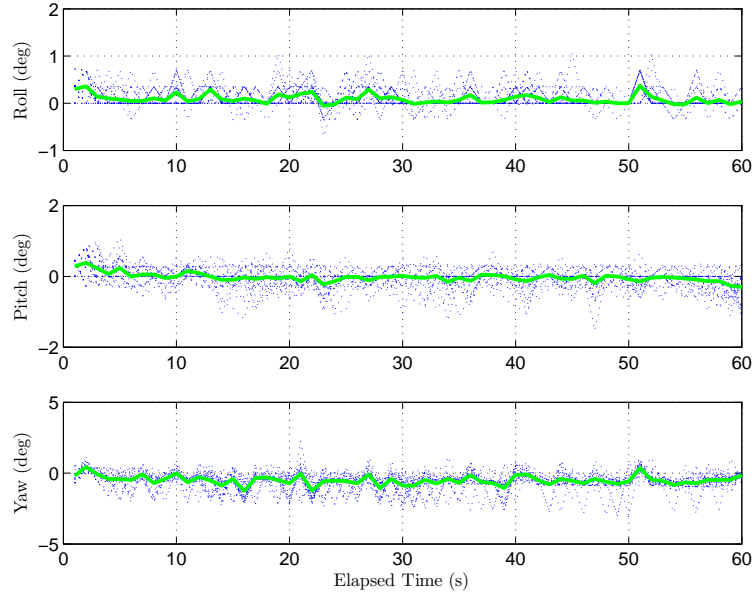


Figure 4.22: Attitude measurement residuals for the Monte Carlo simulation using navigation grade IMU at the pre-contact position. The sample functions are indicated by the blue, dotted lines. The ensemble mean is indicated by the green line.

4.2.1.2 *Tactical Grade IMU.* As a performance comparison, another Monte Carlo simulation of the same trajectory is presented using the tactical grade IMU parameters. First, the free inertial simulation results are shown in Figure 4.23. Secondly, the results for this simulation utilizing the image update are shown in Figures 4.24-4.26.

These figures show similar performance to the navigation grade inertial sensors in the  $y$  and  $z$  axes. The errors in the  $x$  body-frame are very consistent with those in the navigation grade simulation. The measurement residuals are very similar in behavior and magnitude between the tactical and navigation grade sensors. For this particular simulation, either grade of sensor yields accurate relative positioning.

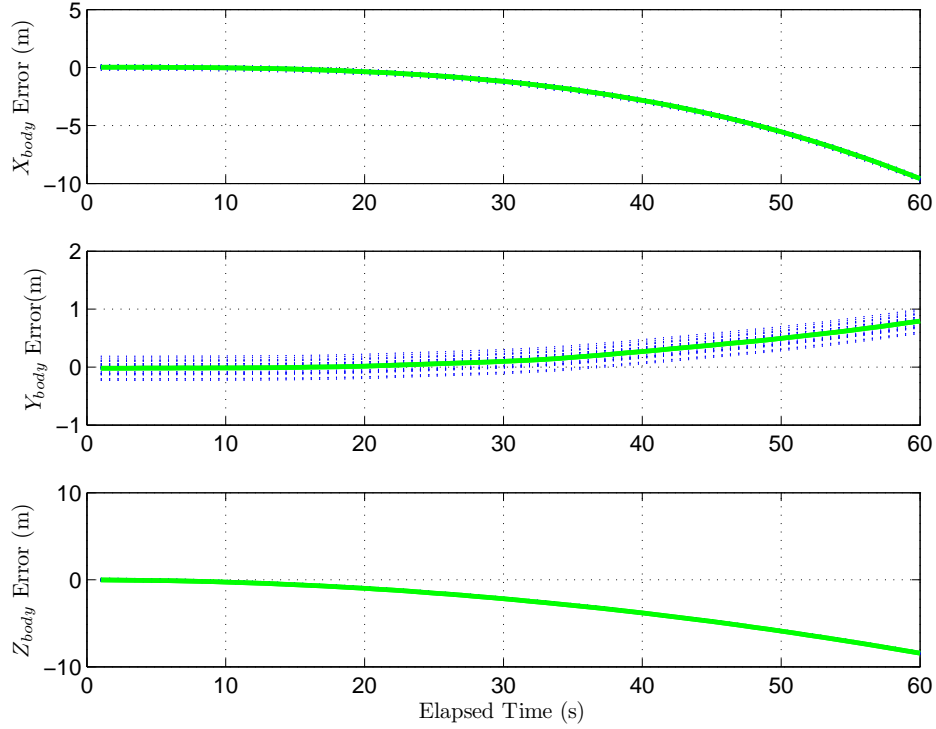


Figure 4.23: The mean relative position error (indicated by the green line) for the Monte Carlo simulation using tactical grade IMU and no image updates, expressed in the wing body frame while at the pre-contact position. The sample functions for the 30 iterations are indicated by the blue, dotted lines.

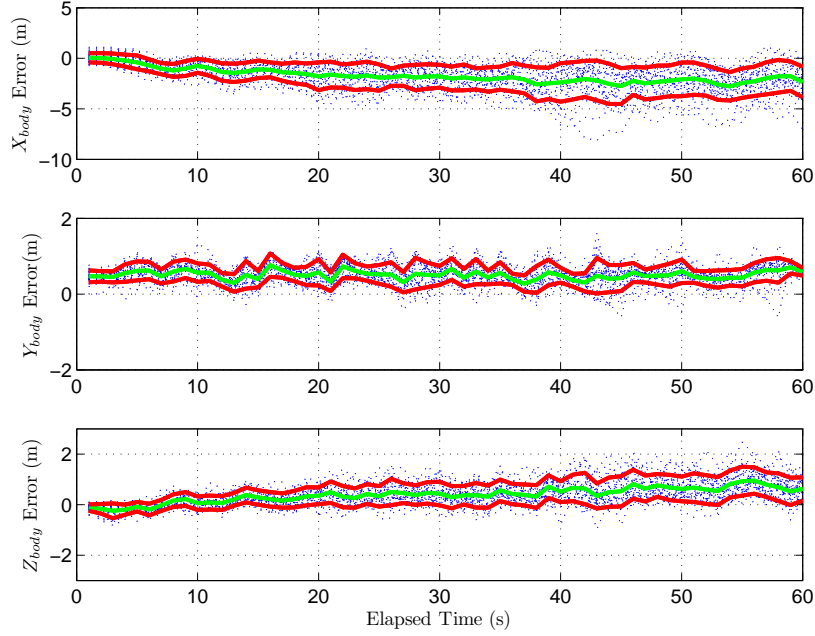


Figure 4.24: Relative position error for the Monte Carlo simulation using tactical grade IMU, expressed in the wing body frame while at the pre-contact position. The relative error sample functions are indicated by the blue, dotted lines. The solid green curve is the ensemble mean, and the red curves indicate the ensemble standard deviation (mean  $\pm 1\sigma$  bounds).

#### 4.2.2 Simple Trajectory at Pre-contact Position with Initial Condition Errors.

In order to test the filter's capability to reject estimation errors, another simulation analyzes significant errors in the initial conditions. In this simulation, the initial position of the lead aircraft contains an intentional, random error of 5 meters ( $1\sigma$ ) in the East direction, as well as a roll error of  $2^\circ$  ( $1\sigma$ ). This is a northward trajectory (just as the previous simulation), so the position errors will appear in the  $y$  axis of the wing body frame. Due to the intentional offset in East position and roll, the initial covariance values for these states are increased to  $5^2 \text{ (m)}^2$  and  $2^2 \text{ (deg)}^2$ , respectively. The remaining initial conditions and covariance values are the same as described in the previous simulation (Table 4.2). This 30 iteration Monte Carlo simulation uses a navigation grade IMU. Figures 4.27-4.29 display the results of this simulation.

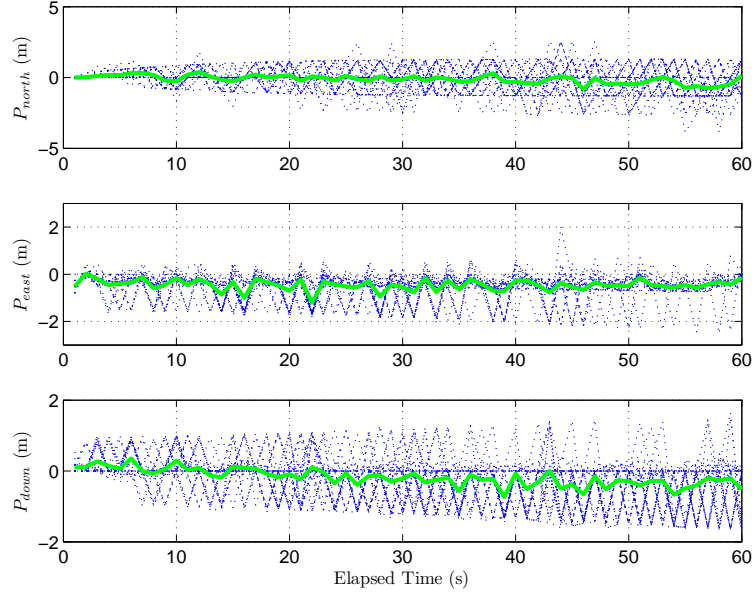


Figure 4.25: Position measurement residuals for the Monte Carlo simulation using tactical grade IMU at the pre-contact position. The sample functions are indicated by the blue, dotted lines. The ensemble mean is indicated by the green line.

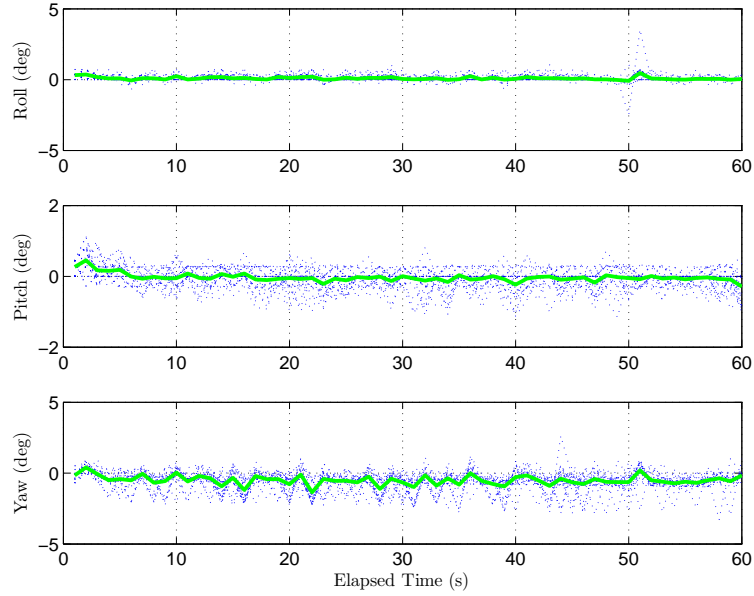


Figure 4.26: Attitude measurement residuals for the Monte Carlo simulation using tactical grade IMU at the pre-contact position. The sample functions are indicated by the blue, dotted lines. The ensemble mean is indicated by the green line.

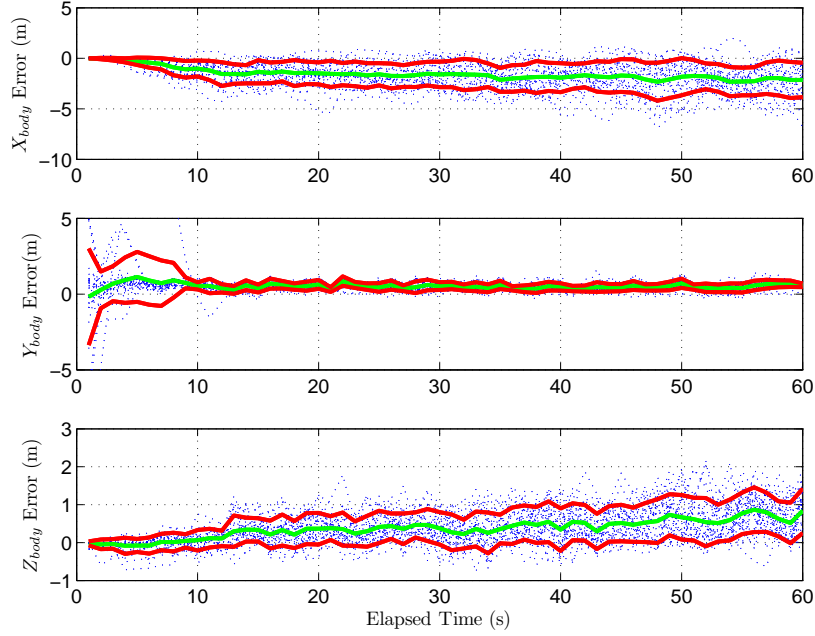


Figure 4.27: Relative position error for the Monte Carlo simulation, expressed in the wing body frame with intentional errors in initial conditions. The relative error sample functions are indicated by the blue, dotted lines. The solid green curve is the ensemble mean, and the red curves indicate the ensemble standard deviation (mean  $\pm 1\sigma$  bounds).

From these figures, the filter demonstrates its capability to recover from initial error estimates. Within 10 seconds, the filter corrected for the  $y$  body error, and the roll measurement residuals were near zero. Again, the largest relative position errors are seen in the  $x$  axis due to the observability issues. After correcting the initial error, the position residuals (Figure 4.28) are less than  $|1 \text{ m}|$  and the attitude residuals (Figure 4.29) never exceed  $|2^\circ|$ . Overall, the error magnitudes are similar to the simulation that was initialized with the true conditions. This is an important feature of the filter's robust operation, since the true initial navigation state will seldom be known when executing a real operation.

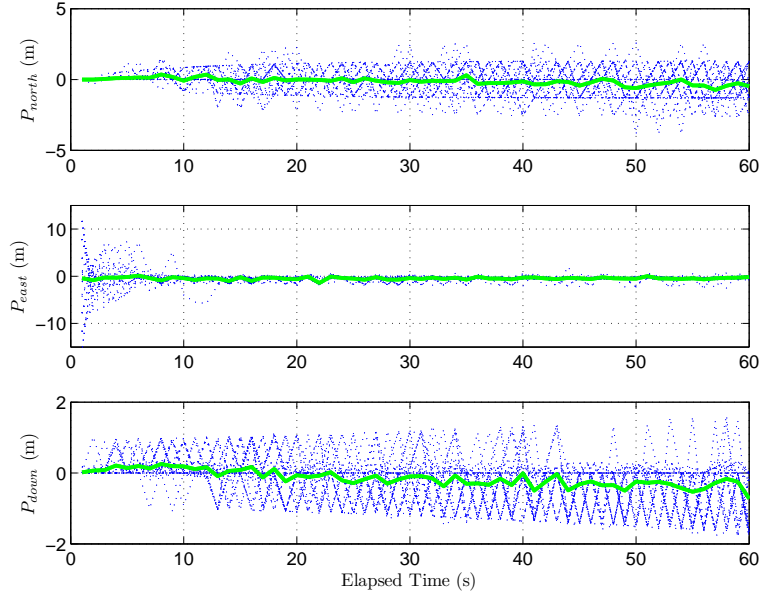


Figure 4.28: Position measurement residuals for the Monte Carlo simulation with intentional errors in initial conditions. The sample functions are indicated by the blue, dotted lines. The ensemble mean is indicated by the green line.

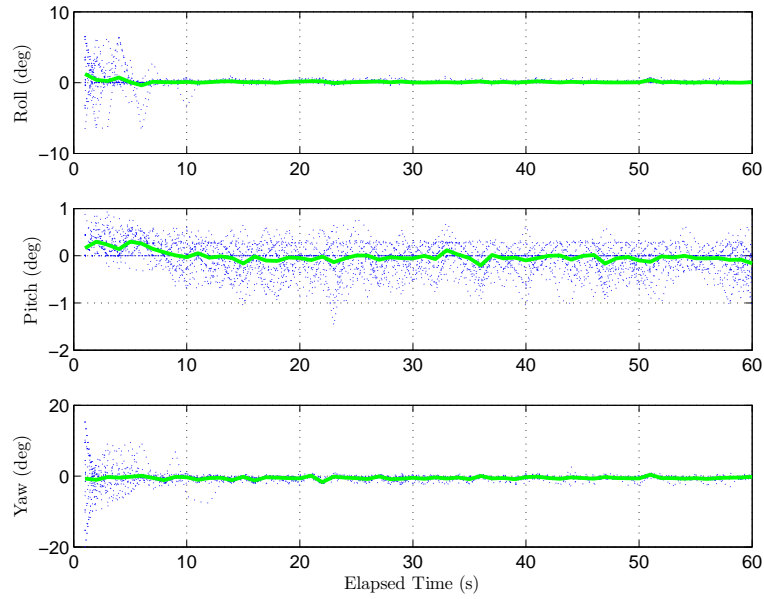


Figure 4.29: Attitude measurement residuals for the Monte Carlo simulation with intentional errors in initial conditions. The sample functions are indicated by the blue, dotted lines. The ensemble mean is indicated by the green line.

*4.2.3 Simple Trajectory at the Contact Position.* The next simulations consider the receiver in the contact position. The filter is initialized with the true initial conditions, as well as the state error covariance values from Table 4.2.

*4.2.3.1 Navigation Grade IMU.* The first simulation considers the navigation grade IMU. Figures 4.30-4.32 display the results of this simulation. The  $x$  axis errors in the wing body frame again show larger magnitudes than the  $y$  and  $z$  axes. Recall from Figure 2.5 that only a portion of the lead aircraft is visible at the contact position. The wing aircraft camera field of view can typically only capture a portion of the fuselage and the wings just beyond the two inboard engines. This narrow view of the tanker makes errors in  $x$  axis translation and pitch difficult to detect.

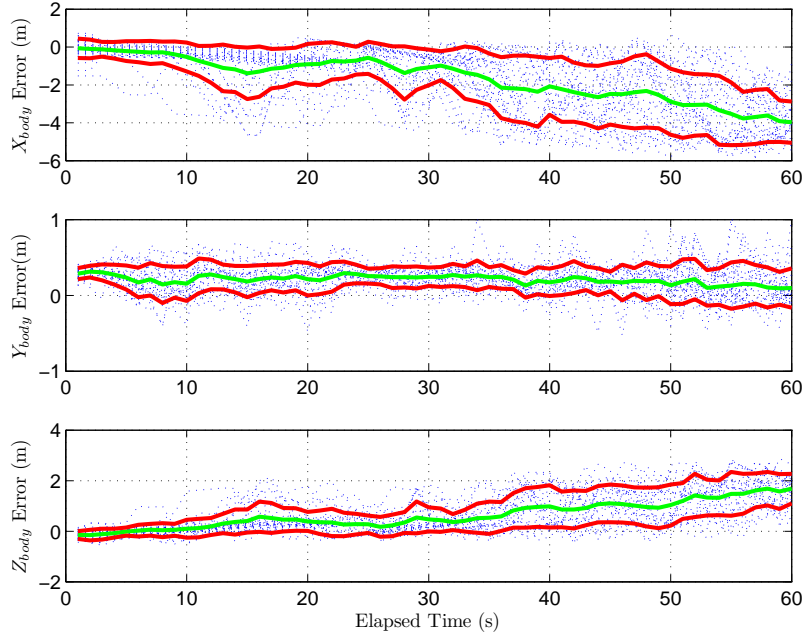


Figure 4.30: Relative position error for the Monte Carlo simulation using navigation grade IMU, expressed in the wing body frame while at the contact position. The relative error sample functions are indicated by the blue, dotted lines. The solid green curve is the ensemble mean, and the red curves indicate the ensemble standard deviation (mean  $\pm 1\sigma$  bounds).



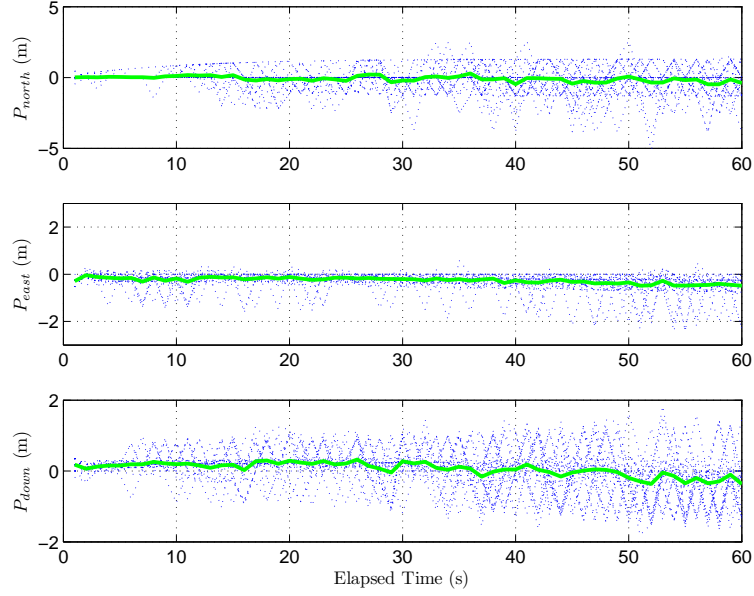


Figure 4.31: Position measurement residuals for the Monte Carlo simulation using navigation grade IMU at the contact position. The sample functions are indicated by the blue, dotted lines. The ensemble mean is indicated by the green line.

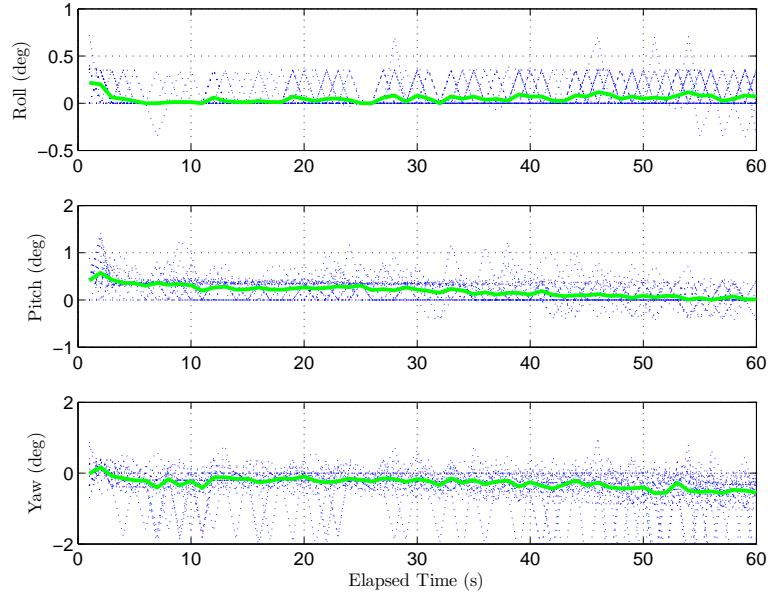


Figure 4.32: Attitude measurement residuals for the simulation using navigation grade IMU at the contact position. The sample functions are indicated by the blue, dotted lines. The ensemble mean is indicated by the green line.

With the limited amount of the lead aircraft in the field of view, these errors would be even more pronounced than in the pre-contact position. Examining the measurement residuals, the magnitudes are very small during the first half of the simulation, then begin to slightly increase in the latter half. This is consistent with the relative position errors of Figure 4.30, where the error and covariance in the  $x$  body axis suddenly increase. The most likely explanation is that a few of the trajectory runs had a bad measurement update at this point in the simulation, consistent with the decreased observability to the  $x$  axis errors.

*4.2.3.2 Tactical Grade IMU.* This same contact position trajectory is simulated using a tactical grade IMU. The results are displayed in Figures 4.33-4.35. Again, the results here are fairly similar to the previous tactical grade simulation. The tactical grade IMU shows the largest magnitude of error in the  $x$  body axis, as expected. The largest errors occur near the end of the simulation (after 50 seconds), indicating that the lower grade inertial sensor does begin to affect performance over time. Ultimately, either choice of sensor can yield a reasonably accurate relative position solution for this simple trajectory, though the navigation grade provides a small performance enhancement.

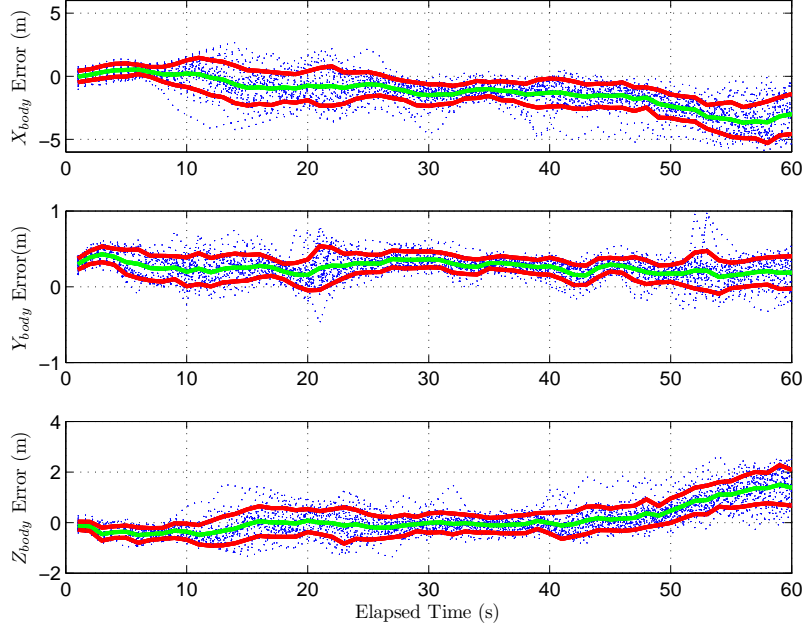


Figure 4.33: Relative position error for the Monte Carlo simulation using a tactical grade IMU, expressed in the wing body frame while at the contact position. The relative error sample functions are indicated by the blue, dotted lines. The solid green curve is the ensemble mean, and the red curves indicate the ensemble standard deviation (mean  $\pm 1\sigma$  bounds).

### 4.3 Summary

This chapter has highlighted the strengths and weaknesses of three image processing techniques. Both the standalone sum-squared difference approach and the approach coupling sum-squared difference with magnitude of gradient give reasonable error detection performance in the simulations.

The simulated trajectories show the validity of using predictive rendering in an image-aided EKF for relative positioning. Though not yet at a maturity level for extremely precise (centimeter level) positioning, this research has demonstrated that the vision-aided approach is a valuable tool for finding an accurate relative navigation solution. Furthermore, this technique would be a robust addition to systems primarily relying on GPS. More on this technique's future maturation and recommendations for continued research are presented in Chapter V.

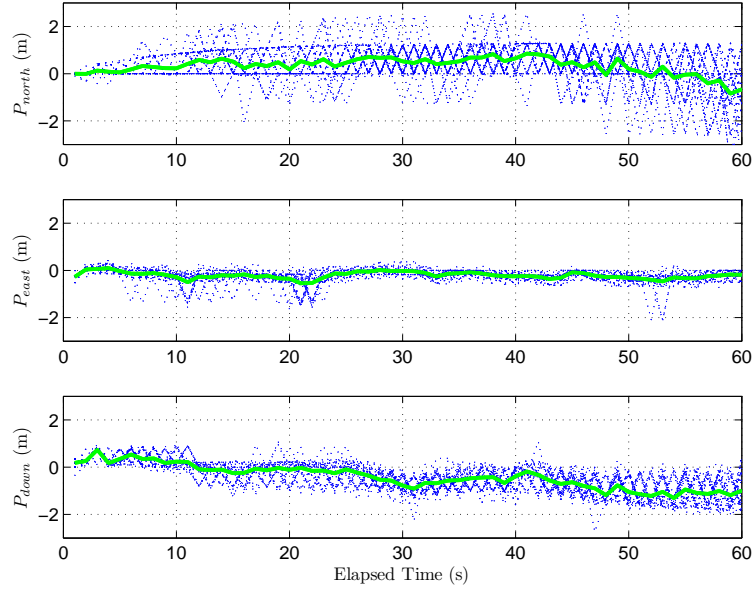


Figure 4.34: Position measurement residuals for the Monte Carlo simulation of straight and level flight using a tactical grade IMU at the contact position. The sample functions are indicated by the blue, dotted lines. The ensemble mean is indicated by the green line.

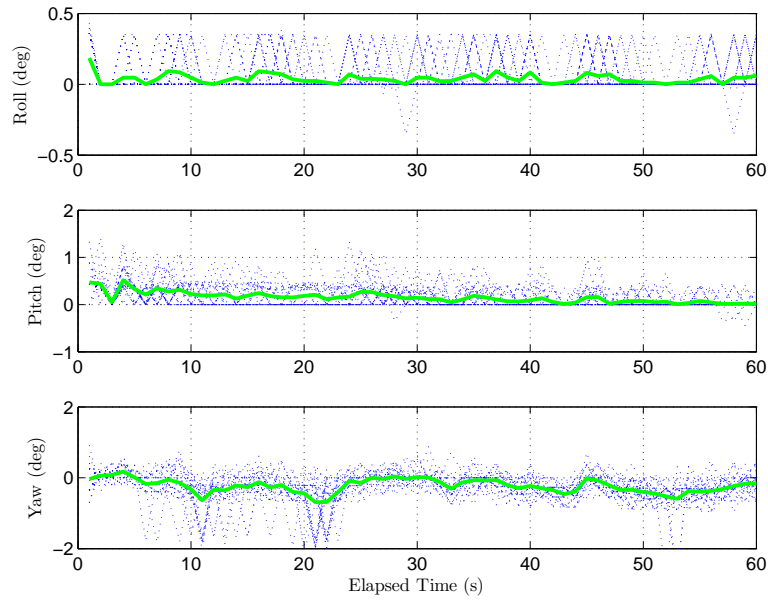


Figure 4.35: Attitude measurement residuals for the Monte Carlo simulation of straight and level flight using a tactical grade IMU at the contact position. The sample functions are indicated by the blue, dotted lines. The ensemble mean is indicated by the green line.

## V. Conclusions and Recommendations

Based on the results from Chapter IV, this chapter will highlight the overall conclusions from this research as well as recommendations for future research related to this topic.

### 5.1 *Research Conclusions*

The image processing analysis verified that the predictive rendering approach is valid for detecting errors between a predicted and true image. Though there are several modes that show observability issues (in particular the mode that couples errors in the  $x$  body frame with errors in pitch), the simulations demonstrated that the image update can still adequately detect residual errors. The approach of first using sum-squared difference for coarse error detection, then transitioning to magnitude of gradient performed the best considering all image processing simulations.

Examining the simulation results, one can see that the system demonstrates reasonably accurate relative positioning during the pre-contact positions. The mean error in relative position is often less than two meters with the navigation grade IMU, with a few  $x$  body frame errors exceeding this threshold. The mean  $y$  and  $z$  estimates are nearly always within one meter of the true value. For the pre-contact position, this level of accuracy is acceptable, since there is no physical contact between aircraft and no danger of collision.

In the case of initial condition errors, the filter rapidly corrects these initial errors and shows similar performance to the previous simulation. The mean  $x$  axis errors are the largest in magnitude, and generally stay less than approximately two meters.

Now examining the contact position, recall from Chapter II the previous research conducted by Ross (Section 2.10.1). He noted that an optical system would need to maintain accuracy to within five feet of the nominal position, thus allowing the boom operator to correct the residual error. Therefore, an accuracy threshold of approximately 1.5 meters is appropriate. The simulations for both the navigation

and tactical grade inertial sensors exceeded this threshold (the mean error deviating to as much as four meters). The current navigation solution alone would not provide accurate enough positioning at the contact position to yield satisfactory and safe operation. Additional refinement is needed to fine tune the accuracy to one meter or better.

## **5.2 Recommendations for Future Research**

*5.2.1 Additional Sensor Analysis.* This research used predictive rendering to generate a measurement update during simple trajectory simulations. Considering the pre-contact and contact positions, both navigation and tactical grade inertial sensors were simulated through various trajectories. These simulations were designed to be completely passive for the operation of the tanker. In other words, the system under test did not rely on any modifications to the tanker aircraft (such as datalink capability or optical markers). Though providing the baseline validation of the predictive rendering approach with a completely passive system, there is more research to investigate with this technique.

Additional sensor parameters (including ranging sensors, datalink between aircraft, and dual GPS updates) should be analyzed to completely characterize the autonomous refueling problem. The use of a datalink providing inertial or GPS position updates would definitely be worth investigating, though this would require tanker fleet modifications. Furthermore, including a ranging sensor on the receiver aircraft could provide additional information to complement the image update function and make it more accurate. An overall sensitivity analysis is needed to truly characterize the most robust, fault-tolerant image-aided system. Also, turning trajectories are typical in aerial refueling maneuvers and should be analyzed using this technique.

*5.2.2 Modification for Real-Time Operation.* The **Matlab**<sup>®</sup> simulation using VRML to render the images is a valid, yet time-consuming approach. There is a definite delay between commanding the rendering parameters and the VR window

actually displaying the rendering, even noticeable to the human eye. In order to mitigate this, other rendering methods should be explored. Specifically, directly using the graphics processing unit to render would likely be the optimal approach to achieve high enough frame rates to render and process many images every second.

This research used an EKF, which generates only one navigation state estimate. Thus, perturbations about this state estimate were needed to determine the residual errors between the predicted and measured relative position and orientation. Rendering each of these perturbations and iteratively adjusting the prediction to find the closest matching image was a lengthy, time-consuming process. For real-time operation, this approach would not be practical. To enhance the speed of the image update, an Unscented Kalman Filter (UKF) should be implemented. With a UKF, a large number of sigma points would be generated instead of a single state estimate. With each sigma point as a prediction (and thus predicted image), the image update could develop some type of hypersurface to characterize how the errors vary with the sigma point spread. Based on this distribution of error values, the update function could determine where the minimum error, and thus optimal state estimate, occurs. The function would still be generating a large number of rendered images, but would not be performing the time-consuming perturbation process. This would allow the image update to capture a truth image from the digital camera, compare the sigma point predictions against the truth, determine the nominal estimate, and pass the nominal back to the filter for propagation to the next measurement. All of this must occur quickly enough to capture truth images from the digital camera at a 1 Hz or possibly 2 Hz rate.

Finally, with a filter and image update configuration that can operate in real-time, a controller is needed to correct the receiver aircraft position based on the navigation state estimate. This could be a simple regulator that drives the error state to zero, thus keeping the receiver aircraft at the nominal pre-contact or contact position, as appropriate.

## Bibliography

1. Department of Defense World Geodetic System 1984 - Its Definition and Relationships with Local Geodetic Systems. In-House Technical Report 8350.2, Defense Mapping Agency, Washington, DC, September 1987.
2. Barfield, A. Finley and Jacob Hinchman. "An Equivalent Model for UAV Automated Aerial Refueling Research". *AIAA Modeling and Simulation Technologies Conference*. Air Force Research Laboratory, Air Vehicles Directorate, Wright-Patterson AFB, OH, August 2005.
3. Bowers, Roshawn and Brian Stolarik. *Station Keeping Flight Test Hybrid System Report*. In-House FA8650-04-2-3450, Northrop Grumman Corporation, El Segundo, CA, September 2006.
4. Fravolini, M.L., G. Campa, and M.R. Napolitano. "Evaluation of Machine Vision Algorithms for Autonomous Aerial Refueling for Unmanned Aerial Vehicles". *Journal of Aerospace Computing, Information, and Communication*, 4:968–985, September 2007.
5. Fravolini, M.L., Antonio Ficola, Giampiero Campa, Marcello Rosario Napolitano, and Brad Seanor. "Modeling and Control Issues for Autonomous Aerial Refueling for UAVs Using a ProbeDrogue Refueling System". *Aerospace Science and Technology*, 8:611–618, June 2004.
6. Groves, Paul D. *Principles of GNSS, Inertial, and Multisensor Integrated Navigation Systems*. Artech House, Boston, MA, 2008.
7. Hansen, Jennifer L., James E. Murray, and Norma V. Campos. "The NASA Dryden AAR Project: A Flight Test Approach to an Aerial Refueling System". *AIAA Atmospheric Flight Mechanics Conference and Exhibit*. NASA Dryden Flight Research Center, Edwards, California, August 2004.
8. Hinchman, Jacob and Daniel Schreiter. "Automated Aerial Refueling". *Presentation to Aerial Refueling Systems Advisory Group (ARSAG)*. Air Force Research Laboratory, Air Vehicles Directorate, Wright-Patterson AFB, OH, April 2007.
9. Khanafseh, Samer A. and Boris Pervan. "Autonomous Airborne Refueling of Unmanned Air Vehicles Using the Global Positioning System". *Journal of Aircraft*, 44(5):1670–1682, September-October 2007.
10. Maybeck, Peter S. *Stochastic Models, Estimation and Control, Vol II*. Academic Press, Inc., Orlando, FL, 1979.
11. Maybeck, Peter S. *Stochastic Models, Estimation and Control, Vol I*. Academic Press, Inc., Orlando, FL, 1994.



12. Orton, Megan. "Air Force remains committed to unmanned aircraft systems". Online, January 2009. Available at <http://www.af.mil/news/story.asp?id=123131324>.
13. Ross, Steven M. *Formation Flight Control for Aerial Refueling*. Master's thesis, Graduate School of Engineering, Air Force Institute of Technology (AETC), Wright-Patterson AFB OH, March 2006. AFIT/GAE/ENY/06-M35.
14. Seanor, Brad, Giampiero Campa, Yu Gu, Marcello Napolitano, Larry Rowe, and Mario Perhinschi. "Formation Flight Test Results for UAV Research Aircraft Models". *AIAA 1st Intelligent Systems Technical Conference*. West Virginia University, Morgantown, West Virginia, September 2004.
15. Spencer, James H. *Optical Tracking for Relative Positioning in Automated Aerial Refueling*. Master's thesis, Graduate School of Engineering, Air Force Institute of Technology (AETC), Wright-Patterson AFB OH, March 2007. AFIT/GE/ENG/07-22.
16. Spinelli, Christopher J. *Development and Testing of a High-Speed Real-Time Kinematic Precise DGPS Positioning System Between Two Aircraft*. Master's thesis, Graduate School of Engineering, Air Force Institute of Technology (AETC), Wright-Patterson AFB OH, September 2006. AFIT/GCD/ENG/06-12.
17. Titterton, David H. and John L. Weston. *Strapdown Inertial Technology*. The Institution of Electrical Engineers, Stevenage, Herts, UK, 2004.
18. Valasek, John, Kiran Gunnam, Jennifer Kimmet, Monish D. Tandale, and John L. Junkins. "Vision-Based Sensor and Navigation System for Autonomous Air Refueling". *Journal of Guidance, Control, and Dynamics*, 28(5):979–989, September-October 2005.
19. Veth, Michael J. *Fusion of Imaging and Inertial Sensors for Navigation*. Ph.D. dissertation, Graduate School of Engineering, Air Force Institute of Technology (AETC), Wright-Patterson AFB OH, August 2006. AFIT/DS/ENG/06-09.
20. Warwick, Graham. "AFRL Advances Autonomous Aerial Refueling". *Aviation Week*, June 2008.
21. Woodman, Oliver J. *An Introduction to Inertial Navigation*. In-House Technical Report 696, University of Cambridge, Cambridge, UK, September 2007.

| <b>REPORT DOCUMENTATION PAGE</b>  |             |  |   |   | <i>Form Approved</i><br><b>OMB No. 0704-0188</b>                         |  |
|---|-------------|--|---|---|--|--|
| The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number. <b>PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.</b>  |             |  |   |   |  |  |
| <b>1. REPORT DATE</b> (DD-MM-YYYY)<br>05-03-2009  |             | <b>2. REPORT TYPE</b><br>Master's Thesis |   |   | <b>3. DATES COVERED</b> (From — To)<br>Sept 2007 — Mar 2009              |  |
| <b>4. TITLE AND SUBTITLE</b><br><br><div style="text-align: center;">Using Predictive Rendering as a Vision-Aided Technique<br/>for Autonomous Aerial Refueling</div>   |             |  |   | <b>5a. CONTRACT NUMBER</b>  |  |  |
|   |             |  |   | <b>5b. GRANT NUMBER</b>   |  |  |
|   |             |  |   | <b>5c. PROGRAM ELEMENT NUMBER</b>   |  |  |
| <b>6. AUTHOR(S)</b><br><br>Weaver, Adam D., Capt, USAF  |             |  |   | <b>5d. PROJECT NUMBER</b><br><br>JON 09-319   |  |  |
|   |             |  |   | <b>5e. TASK NUMBER</b>  |  |  |
|   |             |  |   | <b>5f. WORK UNIT NUMBER</b>   |  |  |
| <b>7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)</b><br>Air Force Institute of Technology<br>Graduate School of Engineering and Management (AFIT/EN)<br>2950 Hobson Way<br>WPAFB OH 45433-7765   |             |  |   |   | <b>8. PERFORMING ORGANIZATION REPORT NUMBER</b><br><br>AFIT/GE/ENG/09-45 |  |
| <b>9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)</b><br>Air Force Research Laboratory Sensors Directorate<br>Attn: Jacob M. Campbell<br>2241 Avionics Circle<br>WPAFB OH 45433-7303<br>((937)255-6127 ext 4154, jacob.campbell@wpafb.af.mil)  |             |  |   |   | <b>10. SPONSOR/MONITOR'S ACRONYM(S)</b><br><br>AFRL/RYRN                 |  |
|   |             |  |   |   | <b>11. SPONSOR/MONITOR'S REPORT NUMBER(S)</b>                            |  |
| <b>12. DISTRIBUTION / AVAILABILITY STATEMENT</b><br><br>Approval for public release; distribution is unlimited.   |             |  |   |   |  |  |
| <b>13. SUPPLEMENTARY NOTES</b>  |             |  |   |   |  |  |
| <b>14. ABSTRACT</b><br>This research effort seeks to characterize a vision-aided approach for an Unmanned Aerial System (UAS) to autonomously determine relative position to another aircraft in a formation, specifically to address the autonomous aerial refueling problem. A system consisting of a monocular digital camera coupled with inertial sensors onboard the UAS is analyzed for feasibility of using this vision-aided approach. A three-dimensional rendering of the tanker aircraft is used to generate predicted images of the tanker. A rigorous error model is developed to model the relative dynamics. To quantify the errors between the predicted and true images, an image update function is developed using perturbation techniques. Based on this residual measurement and the inertial/dynamics propagation, an Extended Kalman Filter (EKF) is used to predict the relative position and orientation of the tanker from the receiver aircraft. The EKF is simulated through various formation positions during typical aerial refueling operations. Various grades of inertial sensors are simulated to analyze the system's ability to accurately and robustly track the relative position between the two aircraft. |             |  |   |   |  |  |
| <b>15. SUBJECT TERMS</b><br><br>autonomous aerial refueling, vision-aided navigation, predictive rendering, image prediction, image comparison  |             |  |   |   |  |  |
| <b>16. SECURITY CLASSIFICATION OF:</b>  |             |  | <b>17. LIMITATION OF ABSTRACT</b><br><br>UU                                   |   | <b>18. NUMBER OF PAGES</b><br><br>90                                     |  |
| a. REPORT   | b. ABSTRACT | c. THIS PAGE                             |   |   |  |  |
| U   | U           | U  | <b>19a. NAME OF RESPONSIBLE PERSON</b><br>Michael J. Veth, Lt Col, USAF (ENG) |   |  |  |
|   |             |  |   | <b>19b. TELEPHONE NUMBER</b> (include area code)<br>(937) 255-3636, ext 4541; michael.veth@afit.edu |  |  |