3-2008

Development and Flight of a Robust Optical-Inertial Navigation System Using Low-Cost Sensors

Michael B. Nielsen

DEVELOPMENT AND FLIGHT TEST OF A ROBUST OPTICAL-INERTIAL NAVIGATION SYSTEM USING LOW-COST SENSORS

THESIS

Michael B. Nielsen, Captain, USAF

AFIT/GE/ENG/08-19

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

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 U.S. Government.

AFIT/GE/ENG/08-19

DEVELOPMENT AND FLIGHT TEST OF A ROBUST OPTICAL-INERTIAL NAVIGATION SYSTEM USING LOW-COST SENSORS

THESIS

Presented to the Faculty

Department of Electrical 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

Michael B. Nielsen, BS

Captain, USAF

March 2008

AFIT/GE/ENG/08-19

DEVELOPMENT AND FLIGHT TEST OF A ROBUST OPTICAL-INERTIAL NAVIGATION SYSTEM USING LOW-COST SENSORS

Michael B. Nielsen, BS

Captain, USAF

Approved:

//SIGNED//	
Dr. John Raquet	Date
Thesis Advisor, Committee Chairman	
//SIGNED//	
Dr. Peter Maybeck	Date
Committee Member	
//SIGNED//	
Dr. Meir Pachter	Date
Committee Member	
//SIGNED//	
Major Michael Veth, USAF	Date
Committee Member	

AFIT/GE/ENG/08-19

Abstract

This research develops and tests a precision navigation algorithm fusing optical and inertial measurements of unknown objects at unknown locations. It provides an alternative to the Global Positioning System (GPS) as a precision navigation source, enabling passive and low-cost navigation in situations where GPS is denied/unavailable.

This paper describes two new contributions. First, a rigorous study of the fundamental nature of optical/inertial navigation is accomplished by examining the observability grammian of the underlying measurement equations. This analysis yields a set of design principles guiding the development of optical/inertial navigation algorithms.

The second contribution of this research is the development and flight test of an optical-inertial navigation system using low-cost and passive sensors (including an inexpensive commercial-grade inertial sensor, which is unsuitable for navigation by itself). This prototype system was built and flight tested at the U.S. Air Force Test Pilot School. The algorithm that was implemented leveraged the design principles described above, and used images from a single camera. It was shown (and explained by the observability analysis) that the system gained significant performance by aiding it with a barometric altimeter and magnetic compass, and by using a digital terrain database (DTED). The (still) low-cost and passive system demonstrated performance comparable to high quality navigation-grade inertial navigation systems, which cost an order of magnitude more than this optical-inertial prototype. The resultant performance of the system tested provides a robust and practical navigation solution for Air Force aircraft.

To my wife and son who have suffered my ambitions, and bore my burdens.

Acknowledgments

I would like to express my sincere appreciation to my faculty advisor, Dr. John Raquet, for his guidance and support throughout the course of this thesis effort. The mentorship I received will prove fruit the rest of my career. I would also like to thank my mentors on the subject matter, Dr Meir Pachter, Dr Peter Maybeck, and Maj Michael Veth for both the insight and experience that they have shared over the past two and a half years.

I would like to express my thanks to the faculty and staff at the United States Air Force Test Pilot School for their diligent work, training me to be a test pilot in America's Air Force and making this thesis possible.

I am obliged by tradition to thank the 494th Mighty Black Panthers, the finest, baddest, meanest fighter squadron for molding me into the officer and warrior that I am, and for continually making America's enemies bleed red in my absence.

Most importantly I have to thank my beautiful wife, precocious son, and merciful Savior for making this life and this pursuit worth living.

Michael B. Nielsen

Table of Contents

Page

Abstract iv
Acknowledgments vi
Table of Contents vii
List of Figures xiv
List of Abbreviations xviii
I. Introduction 1
1.1 Research Motivation 1
1.1.1 Importance of Precision Navigation. 2
1.1.2 What is Inertial Navigation? 2
1.1.3 Where is Inertial Navigation Used? 3
1.1.4 Problems with Inertial Navigation Systems 3
1.1.5 Aiding a Drifting Inertial Navigation System 4
1.1.6 Inertial Navigation Aiding 4
1.1.7 Global Positioning System as an Aiding Source 5
1.1.8 Using Imagery to Aid Inertial Navigation 6
1.1.9 Low Cost Inertial Navigation 8
1.2 Problem Statement 9
1.3 Scope and Assumptions 9
1.4 Related Research 10
1.4.1 Vision to Generate Relative Navigation Commands 10
1.4.2 Image Aiding with A Priori Information 11

1.4.3 Image Aiding with No A Priori Information 12
1.4.4 Image Aiding an EGI during GPS Drop Outs 13
1.4.5 Stochastic Image Correspondence 14
1.4.6 Experimental Validation – Peeping Talon 15
1.4.7 Genesis of this Research 16
1.5 Methodology 17
II. Mathematical Background 19
2.1 Mathematical Notation 19
2.2 Reference Frames 21
2.2.1 True Inertial Frame (I-frame) 21
2.2.2 Earth-Centered Inertial Frame (i-frame) 21
2.2.3 Earth-Centered Earth-Fixed Frame (e-frame or ECEF frame) 21
2.2.4 Navigation Frame (n-frame) 21
2.2.5 Body Frame (b-frame) 22
2.2.6 Camera Frame (c-frame) 22
2.3 Transforms Between Reference Frames 25
2.4 Physical System Modeling 28
2.4.1 Linear Systems 28
2.4.2 Non-Linear Systems 29
2.5 Inertial Navigation 33
2.5.1 Shape of the Earth 33
2.5.2 Gravitational Model 34

2.5.3 Accelerometer Sensor Model....36
2.5.4 Gyroscope Sensor Model....37
2.5.5 Strapdown INS mechanization....38
2.5.5.1 Attitude Mechanization....39
2.5.5.2 Position Mechanization....43
2.5.5.3 Velocity Mechanization....45
2.5.5.4 Navigation State....46
2.5.6 Error States....46
2.6 Kalman Filtering....54
2.6.1 Linear Stochastic System Model....54
2.6.2 Linear Measurement Model....56
2.6.3 Linear Kalman Filter....57
2.6.3.1 Kalman Filter Cycle....57
2.6.3.2 Propagation Equations....58
2.6.3.3 Update Equations....59
2.6.4 Non-Linear Stochastic System Model....59
2.6.5 Non-Linear Measurement Model....60
2.6.6 Linearized Model....60
2.6.7 Extended Kalman Filter....62
2.6.7.1 EKF Propagation Equations....62
2.6.7.2 EKF Update Equations....62
2.6.8 Extended Kalman Filter and INS Integration....63

2.6.9 Implicit Measurement Equation....64
2.7 Imaging....65
2.7.1 Optical Projection Theory....65
2.7.2 Feature Tracking....73
2.8 Geometric Dilution of Precision....75
2.9 System Observability....77
2.10 Summary....79
III. Developing the SLAAMR Algorithm....80
3.1 Basic EKF System Model....83
3.1.1 Error State Propagation Model....83
3.1.2 Linearized Measurement Model....85
3.1.3 Initial Landmark Position Estimation....85
3.1.3.1 Third Party Survey....86
3.1.3.2 Self-Survey or Auto-location....86
3.2 Feature Tracking Measurement Model....90
3.2.1 Feature Generation....90
3.2.2 Feature Matching....91
3.2.3 Image-Only Measurement Equation....95
3.2.4 Augmented State Vector for Landmark Position Estimation....99
3.2.5 Linearized Image-Only Measurement Equation....100
3.2.6 Tracking Many Unknown Landmarks....107
3.2.7 Tracking Known Landmarks....109

3.3 Image-Only Measurement Observability 111
3.3.1 Surveyed Landmark Tracking Observability 112
3.3.1.1 Non-Dimensional Study of Aiding Strength 121
3.3.1.2 V over h Influence on Aiding Strength 128
3.3.1.3 Look Angle Influence on Aiding Strength 135
3.3.1.4 Scene Geometry Influence on Aiding Strength 139
3.3.2 Alternative Measurements to Augment Observability 139
3.3.3 Estimated Landmark Position Tracking Observability 143
3.3.3.1 Landmark of Opportunity Tracking 145
3.3.3.2 Using DTED to Constrain Errors Further 147
3.3.3.3 Surveyed Landmarks with Augmented State Vector 149
3.4 Self-Surveying Landmarks of Opportunity 153
3.4.1 Building a Map 154
3.4.2 Constraining Drift with Brute Force 155
3.5 Establishing Initial Conditions 157
3.5.1 Far Guess Landmark Initial Position Estimation 159
3.5.2 Average Height Landmark Initial Position Estimation 162
3.5.3 DTED Intersection Initial Landmark Position Estimation 163
3.5.4 Replacing Landmark Tracks 165
3.6 Principles of Design 167
3.7 SLAAMR Algorithm Design 168
3.8 Design Summary 170

IV Flight Test and Experimental Results....171
4.1 Simulation....172
4.1.1 Simulated Images....172
4.1.2 Simulated Inertial Measurements....174
4.2 Test Item Description....175
4.2.1 High Resolution Digital Camera....175
4.2.2 Inertial Measurement Unit....177
4.2.3 Truth Reference Source....179
4.2.4 Sensor Integration....179
4.2.5 Test Aircraft....182
4.2.6 Sensor Harness....184
4.3 Test Item Calibration and Validation....187
4.3.1 Camera Field-of-View....187
4.3.2 Pinhole Camera Model Calibration and Results....188
4.3.3 Sensor Boresight....194
4.3.4 Feature Generation....195
4.3.5 Feature Persistence....196
4.3.5.1 Descriptor Handoff....201
4.3.5.2 Non-Causal Look-Ahead....201
4.4 Flight Plan....202
4.5 Results....204
4.5.1 Effect of Surveyed Landmarks....206

4.5.2 Full Observability with Only Two Tracks 215
4.5.3 Self-Survey 216
4.5.4 Navigation with Zero A Priori Information 223
4.5.5 Effect of Additional, Non-Image Measurements 228
4.5.6 Effect of Number of Landmark Tracks 231
4.5.7 Effect of Camera Look Angle 233
4.5.8 Effect of V over h 236
4.5.9 Partial Survey Tracking 242
4.5.10 Landing Profile Performance 245
4.5.11 Capstone Design – Long Duration Performance 250
4.5.12 Ad Hoc Techniques That Did Not Work 254
4.6 Summary 257
V. Conclusions and Recommendations 258
5.1 Conclusions 258
5.2 Recommendations for Future Work 266
5.3 Summary 267
Bibliography 269
Vita 273

List of Figures

Page

Figure 2.1: Earth-Centered Inertial, Earth-Centered Earth-Fixed, and Navigation Frame 23

Figure 2.2: Aircraft body frame illustration. 23

Figure 2.3: Camera frame illustration. 24

Figure 2.4: Expressing vectors in different frames. 25

Figure 2.5: Effects of centripetal force on the local gravitational field. 35

Figure 2.6: Kalman filter cycle. 57

Figure 2.7: Extended Kalman Filter and INS integration. 63

Figure 2.8: Thin lens camera model. 66

Figure 2.9: Pinhole camera model. 67

Figure 2.10: Image projection model. 68

Figure 2.11: Camera image array. 69

Figure 3.1: Tracking landmarks yields error vectors correlated with INS drift. 82

Figure 3.2: Flight test image with overlaid SIFT© generated features. 91

Figure 3.3: Representative descriptor values versus their element location. 92

Figure 3.4: Constrained feature search volume. 94

Figure 3.5: Unobservable dimensions for two landmark tracks. 121

Figure 3.6: Flight in the vertical plane. 123

Figure 3.7: GDOP vs dwell time in the non-dimensional case study 126

Figure 3.8: Observability grammian matrix condition number vs dwell time in the non-dimensional case study. 127

Figure 3.11: Rear facing camera geometric advantage. 138

Figure 3.12: Iterative map building and aiding. 156

Figure 3.13: *Far Guess* initial landmark position estimation. 159

Figure 3.14: *Average Height* landmark position estimation. 162

Figure 3.15: DTED intersection initial landmark position estimation. 163

Figure 4.1: High resolution digital camera – PL-A471. 176

Figure 4.2: Inertial Measurement Unit – MIDG II INS/GPS. 178

Figure 4.3: System under test block diagram. 181

Figure 4.4: Test aircraft, C-12C Huron. 182

Figure 4.5: Crew positions and test equipment. 183

Figure 4.6: Installed Sensors. 184

Figure 4.7: Sensor harness. 186

Figure 4.8: The effects of the pinhole camera calibration and transform. 190

Figure 4.9: Tracking surveyed landmark to generate calibration error. 192

Figure 4.10: Errors induced by optics and pinhole camera calibration. 193

Figure 4.11: Errors induced by optics and pinhole camera calibration (magnitude). 194

Figure 4.12: Sequential, side-by-side urban images marked with features and correlated matches. 197

Figure 4.13: Frame-to-frame Correlation Coefficients. 198

Figure 4.14: Correlation coefficients with increasing frame interval spacing. 200

Figure 4.15: Flight path flown over Palmdale and Lancaster, CA. 203

Figure 4.16: Tracking surveyed landmarks with REARWARD looking camera..... 207

Figure 4.17: Tracking surveyed landmarks with DOWNWARD looking camera..... 208

Figure 4.18: Tracking surveyed landmarks with FORWARD looking camera..... 209

Figure 4.19: Self-survey process using the *Far Guess* technique..... 216

Figure 4.20: Self-survey of landmarks..... 217

Figure 4.21: Elevation error for a self-survey of landmarks. Landmark elevation errors are sorted in order of magnitude for illustration purposes. 218

Figure 4.22: Effect of camera resolution and tracking angle on self-survey performance. 220

Figure 4.23: Effect of frame rate per degree of tracking angle on self-survey performance..... 222

Figure 4.24: Zero A Priori Knowledge Aiding Performance. 225

Figure 4.25: Simulated Zero A Priori Knowledge Tracking Performance..... 227

Figure 4.26: Effect of additional, non-image measurements..... 229

Figure 4.27: Effect of tracking many landmarks. 232

Figure 4.28: Effect of camera look angle..... 235

Figure 4.29: Effects of V over h on navigation performance. 238

Figure 4.30: Soldier based SLAAMR system..... 240

Figure 4.31: Simulated soldier based SLAAMR performance..... 241

Figure 4.32: Partial survey navigation performance..... 244

Figure 4.33: Landing profile performance using surveyed landmarks..... 247

Figure 4.34: Long duration flight using altitude, heading and DTED measurements... 252

Figure 4.35: Long duration flight using altitude and DTED measurements; no heading measurements used..... 253

List of Abbreviations

Page

AFB	Air Force Base	2
AFIT	Air Force Institute of Technology	1
CCD	Charged-Coupled Device	10
DCM	Direction Cosine Matrix	20
DGPS	Differential Global Positioning System	10
ECEF	Earth-Centered Earth Fixed	21
EKF	Extended Kalman Filter	19
FLIR	Forward Looking Infra-red Radar	5
GDOP	Geometric Dilution of Precision	75
GPS	Global Positioning System	1
IMU	Inertial Measurement Unit	16
INS	Inertial Navigation System	1
JDAM	Joint Direct Attack Munition	3
LOO	Landmark of Opportunity	14
MEMS	Micro-Electrical Mechanical System	1
NED	North, East and Down	44
SLAM	Simultaneous Location and Mapping	7
SLAAMR	Simultaneous Location, Aiding and Mapping Recursively	79
SUT	System Under Test	18
TACAN	Tactical Air Navigation	4

TPS	Test Pilot School	2
TSPI	Time Space Position Information	173
USAF	United States Air Force	2
WGS84	World Geodetic System of 1984	33
ZAPI	Zero A Priori Information	223

(Page Left Blank)

DEVELOPMENT AND FLIGHT TEST OF A ROBUST OPTICAL-INERTIAL NAVIGATION SYSTEM USING LOW-COST SENSORS

I. Introduction

This thesis details the development and experimental testing of an airborne navigation system fusing optical and inertial sensors. The specific focus is on military utility with representative flight environments and equipment. This work is a follow-on to another research effort at the Air Force Institute of Technology (AFIT), employing a similar navigation strategy [27]. Optical inertial navigation is part of a concerted effort by the Advanced Navigation Technology laboratory to develop precision navigation technology in the absence of the Global Positioning System (GPS), since GPS can be jammed by a potential adversary, even one without much sophistication.

1.1 Research Motivation

This research will attempt to characterize inertial navigation aiding with the use of video feature tracking and a low-cost micro-machined electromechanical system (MEMS) based inertial navigation system (INS). This characterization will be used to design a practical algorithm that provides robust precision navigation in dynamic environments for military application.

Simulation will be used to draw conclusions and enhance design. Finally, the theory and algorithm will be evaluated with experimental data collected as a flight test

program at the United States Air Force (USAF) Test Pilot School (TPS), Edwards Air Force Base (AFB), CA.

The United States military holds a significant advantage on the battlefield, in part due to precision navigation technologies. Precision navigation gives commanders precise and accurate information on the battlefield, and allows efficient and effective application of force, minimizing collateral damage. This force multiplying competency has been increasingly dependent on a single technology, exposing a potential Achilles heel. The military's dependence on the Global Positioning System (GPS) for precision navigation demands exploration into alternatives. Specifically, it demands alternatives that will marginalize the impact of GPS, and expand the environment in which a precision navigation advantage can be maintained.

1.1.1 Importance of Precision Navigation. Navigation is a critical war fighting competency, and its importance only increases as technology advances. History provides many examples of the importance of navigation and target location; victory favors the side that can determine these most accurately. USAF tactics are becoming more and more dependent on GPS quality precision navigation, creating a need to protect this capability. As navigation technology advances, the United States must continually re-evaluate the weaknesses that dependence on this new technology exposes. The ability to navigate, while denying the enemy the same, is critical to national defense. In the past, inertial navigation dominated in aviation, but has lost favor to GPS in recent years.

1.1.2 What is Inertial Navigation? Inertial navigation is the art of determining one's position, and attitude, through the use of inertial sensors. Inertial Navigation

Systems (INSs) use accelerometers and gyroscopes to determine a body's acceleration and angular rotation rate. These measurements are integrated to determine position, velocity, and attitude. The passive nature of these sensors makes them extremely resistant to external denial techniques such as jamming—a very attractive property for military applications.

1.1.3 Where is Inertial Navigation Used? Inertial Navigation Systems are critical components onboard every United States Air Force (USAF) combatant aircraft, and many of its weapons systems. Promise of high precision navigation has led to many more uses of INS in recent years. Weapons that used to be guided by gravity alone are now embedded with inertial guidance systems (WCMD, GBU-24, GBU-15, AGM-130 [35]). More modern systems blend INS and GPS. The Joint Direct Attack Munition (JDAM), favored by theatre commanders, is a prime example [30]. Unmanned Aerial Vehicles (UAVs) require an INS to navigate and control flight path. Individual soldiers are being equipped with navigation systems to help prevent fratricide and increase combat effectiveness. INSs can play a critical role in that area as well.

1.1.4 Problems with Inertial Navigation Systems. In order to determine position and attitude, accelerations and angular rates must be integrated. Current inertial sensors are very precise, but not perfect. Unfortunately, the integration of small inertial sensor errors causes the navigation precision to decay over time. This property of an INS requires external aiding to resolve. Additionally, the high quality sensors needed to achieve a high level of precision are very expensive. This poses an economics problem

when attempting to outfit a single soldier, inexpensive UAV, or disposable weapon with a precise navigation system.

1.1.5 Aiding a Drifting Inertial Navigation System. Due to INS integration errors, INS systems become more and more unreliable as time goes on. This requires external aiding in order to bound error growth. The errors cannot be eliminated, but they can be kept from diverging completely. The concept of aiding entails taking a measurement of the outside world, and comparing it to what the INS predicts the measurement to be. Through stochastic modeling and estimation, the INS can determine a *best estimate* of the error in its navigation solution, and then remove it.

1.1.6 Inertial Navigation Aiding. Typically, active sensors, such as Doppler radar, synthetic aperture radar maps and lasers, have been used to accomplish INS aiding [25]. These active sensors are subject to denial techniques and can alert the enemy to one's position. For this reason, active sensors are undesirable for low observable aircraft or for those entering environments hostile to such sensors. A passive sensor is one that can measure the outside world without emitting any radiation itself. Passive external updates maintain the stealth characteristics desirable on many platforms, and are much more resistant to denial. Barometric altimeters and optical/infra-red cameras are passive sensors. Beacon-based navigation systems, such as Tactical Air Navigation (TACAN) and the Global Positioning System, are also passive, since the transmitted signal does not originate from the aircraft, but they are susceptible to denial. Aiding an INS with video imagery provides a passive, jam-resistant solution to INS aiding. Below is a list of current INS aiding devices fielded in the USAF fleet:

- Passive
    - Barometric Altimeter
    - Forward Looking Infrared (FLIR)
    - Visual over-fly updates by the aircrew
    - Star Trackers
- Beacon Based
    - Global Positioning System (GPS)
    - Tactical Air Navigation (TACAN)
    - Long-Range Navigation (LORAN)
- Active
    - Radar Altimeter
    - Synthetic Aperture Radar
    - Laser/Radar Ranging
    - Doppler Radar

1.1.7 Global Positioning System as an Aiding Source. The advent of the GPS has provided an inexpensive, yet highly beneficial method of aiding existing INS platforms [1][7]. It maintains many of the passive aiding benefits because the aircraft need not emit any radiation; the GPS satellites transmit all the signals. Unfortunately, the signal is quite susceptible to denial. In recent years, GPS has become the preferred method of aiding INS onboard USAF tactical aircraft and weapon systems. In an effort to standardize aviation navigation technology across the three services, the H-764G Embedded GPS/Inertial Navigation System (EGI) was chosen to outfit US military aircraft [30]. GPS provides a highly precise position update to the INS at a 1-2 Hz rate. This type of position update benefits the position solution of the INS as well as the attitude and velocity. EGI systems provide a very reliable and precise navigation solution, but can prove to be a liability if GPS is denied in a particular theatre. Many cornerstone systems that rely on precise relative navigation (JDAM and LINK16 for example) would be rendered less effective. Certain environments deny GPS by their very

geometry; buildings and tree cover block the signal. An aided INS could fill the need where GPS falls short. Reliance on GPS as the sole method of INS update, threatens the ability to navigate in hostile environments.

1.1.8 Using Imagery to Aid Inertial Navigation. The concept of aiding an INS with imagery is quite natural. Our own brains do it every day. The inner ear contains semicircular canals that act like rate gyros, detecting rotation. The '*seat of the pants*' sensation allows humans to detect accelerations. Granted, this INS is very poor. One needs only to close his (or her) eyes for a short period while riding in a car before he or she is unsure of where he is or which way hi is facing. If that person opens his eyes, he can blend the images he sees with the sensations he feels to keep track of where he is going and which way he is facing. Why the need for the inertial sensors, though? Assume that same person was sitting in a bus, looking out the window at the side of another bus parked next to him. If one bus started to move, the person couldn't be sure if their bus was moving forward or the other bus backward. The inertial sensors in the body resolve that ambiguity. If the person saw the other bus move relative to himself, but felt no acceleration, he knows with some certainty, what is happening.

The previous example was overly simplified, but the extension into six degree of freedom motion is where image aided INS lives. An aircraft moving along with an INS will have some idea of where it is going, and where it is facing. A camera allows the aircraft to track stationary landmarks on the ground. The aircraft knows the landmarks are stationary, and based on its best guess of its own motion, can estimate where it will see the landmarks in the next camera frame. The difference between where the aircraft

sees the successive object images and where it *expects* to see them holds the key to determining the error in the INS. For example, if the aircraft thinks it's rolling, but the horizon doesn't change, there must be an error in the attitude solution. If the aircraft thinks it is moving very fast, but the objects pass by slowly, there is an error in the velocity solution. This process of Simultaneous Location And Mapping (SLAM) has been proposed in the literature [4][5][6][11], and is the basis for the research herein.

If so much information can be gleaned from the imagery, it's fair to ask why the INS is needed at all. Firstly, the INS is needed to predict where to look for the objects in the next frame. This reduces the computational burden of search for a match. Secondly, images are not perfect. They contain errors, the tracking algorithms induce errors, and two-dimensional images are limited in what information can be reliably obtained about a three-dimensional world. Kalman filtering allows the input from an error-prone INS to complement the input from imperfect image-based estimations. An INS drifts over time, but gives precise rate and rotation information. Tracked landmarks do not drift over time, but still images provide poor rate information, complementing the INS. The resultant navigation solution builds upon the strengths of both components, and diminishes the weaknesses. The result is an aided INS output with bounded error characteristics.

Image aiding an INS promises to provide GPS quality precision. It is true that the benefit of tracking a single landmark does not compare with the precision of a GPS solution. However, image aiding allows the aircraft to track many landmarks and take updates at the frame rate of the camera being used. This is dependent on the processing

rate of the computer, of course, but as computers become faster, this technology becomes more beneficial.

1.1.9 Low Cost Inertial Navigation. Inexpensive systems like UAVs and the JDAM employ inexpensive INS units [30]. Low-cost inertial sensors are pseudonymous with low quality. These systems prove to be very unstable and require much external aiding to be of any use. In recent years, even lower-cost micro-machined electromechanical systems (MEMS) have been developed [23]. They perform even more poorly without external aiding, but are very appropriately priced for such applications. GPS has successfully been used to aid a MEMS INS, delivering acceptable performance for a UAV application [3]. Unfortunately, denial of GPS would render MEMS INS completely useless. A typical EGI system has slow enough drift characteristics that it can withstand rather long GPS outages. A MEMS unit could diverge in about a minute or less, causing a UAV to crash, or a JDAM to miss. Image aiding provides a passive, robust solution and has proven to provide near-GPS level precision [1]. Applying image aiding to a MEMS unit will give precision at a fraction of the cost of a navigation-grade INS, and without the vulnerabilities of an EGI. This research aims to extract equivalent benefit from much lower-cost, and therefore, more appropriate INS systems for UAVs and disposable weapons.

1.2 Problem Statement

Previous work on optical inertial navigation demonstrated its potential, but leaves many questions unanswered. These questions include: what are the true nature and practical limitations of such a system in a full-scale, operationally representative environment? This research will develop a rigorous theoretical observability analysis to characterize the nature of optical aiding in order to design a robust algorithm. Simulation will be used to draw conclusions and enhance design. Flight test of a monocular system incorporating a low-cost MEMS INS will be used to evaluate the theory. The effects of using pre-surveyed landmarks, unknown landmarks, and a mixture of both will be evaluated as well.

1.3 Scope and Assumptions

Much research and study has been done in the field of optical-inertial navigation (as described in the next Section). This research focuses on the single-camera monocular case. Additionally, the flight environment tested will be representative of real-world aircraft, and not limited to small-scale models. A rigorous mathematical study of the influencing factors affecting performance will be accomplished and validated with test data. The end product goal is a fieldable algorithm that accounts for practical system limitations and does not rely on artificial assumptions.

This research does not explore the use of active sensors, nor purpose-built landmarks which have proven to solve many of the image processing challenges [1]. The

image processing side of the problem is not the subject of this research, but is accomplished with a readily available feature generation algorithm. The image processing is not perfect, but practical techniques to improve it for this application are addressed. The research herein concentrates on the navigation aiding component of the problem, specifically as it pertains to contemporary aviation.

1.4 Related Research

The idea of using imagery to aid navigation is nothing new. This research will follow on a previous master's thesis [6], and recent PhD work [26] accomplished at AFIT. Much work has also been done in private industry in this particular field [4][23]. The research herein will explore a specific corner of this vast field of study. The following is a survey of work related to image aided navigation. This section aims to compare and contrast the current state of the art with the original research represented in this thesis. The goal is to establish the validity of the work herein as well as distinguish it as original and noteworthy.

1.4.1 Vision to Generate Relative Navigation Commands. One successful approach to vision-based navigation involves using video to generate relative navigation commands. These commands are fed into a GPS-aided INS for the purposes of landing a UAV. The AVATAR project is a radio controlled helicopter equipped with a Differential GPS (DGPS) onboard to provide absolute position of the aircraft [24]. It uses a CCD camera to find a helipad and land. The helipad is of a known shape and size and made to have a high contrast to the surrounding environment. The high contrast nature of the

helipad allows the AVATAR to filter out the surroundings and calculate its relative position. This research has proven very successful, but differs from the nature of the research herein in many ways. Firstly, the AVATAR relies on DGPS to solve its own position and attitude; the vision algorithms provide nothing in the way of aiding the INS. Additionally, the helipad is a single landmark of known dimensions, simplifying the problem of determining the relative position to the landing zone. Lastly, the aircraft used was a helicopter. The research herein will track many unknown landmarks, with little to no a priori information. The landmarks will aid the INS precision and not be used directly for relative guidance commands. Additionally, the aircraft to be used will be fixed-wing, exhibiting vastly different flight dynamics from those of a helicopter.

1.4.2 Image Aiding with A Priori Information. A group from the Georgia Institute of Technology used a small helicopter to implement a vision-aided INS [1]. This group put an INS on a small GTMax helicopter and aided it with video output from a body-mounted camera [8]. Through the use of an extended Kalman filter, the INS position and attitude were aided without the need of GPS. Simulations were very promising; however, the experiment lacked a truly robust solution. The simulation tracked a single, vertical rectangle of known dimensions and position. This a priori information simplified the problem of updating the onboard INS, but the algorithms are essentially the same as will be discussed herein.

Members of this same group explored using vision as a means of determining position relative to a runway [1]. Beacons were placed on the four corners of a runway of known dimension. The measurements of these beacons (as seen in the video) were used

as the sole inputs to a Kalman filter determining position and attitude. The simulation results indicate that vision alone may be an acceptable solution for the UAV landing problem. These results using a single set of known landmarks (four corners of a rectangle) are promising. The research herein will explore tracking many more landmarks with and without a priori information (including a landing task), but the Georgia Institute of Technology study adds validation to the idea of extracting navigation information from imagery.

1.4.3 Image Aiding with No A Priori Information. A process called Simultaneous Localization And Mapping (SLAM) has been recently researched at the University of Sydney, Australia [10][5]. This process uses video streams to "*map*" terrain according to landmarks of interest. These landmarks are described as "features… that can be consistently and reliably observed" by the aircraft. No a priori information is known about the location of the landmarks. However, the image processing and feature tracking problem is aided with knowledge about the feature's appearance. By tracking the observed landmark position in sequential video frames and comparing with estimated positions through an extended Kalman filter, a 3-dimentional map can be made of the surrounding terrain. These tracked landmarks can then be used to update an INS in position and attitude. Simulation showed impressive results; aircraft were able to obtain near-GPS quality positions and attitudes while flying a SLAM profile circuit. In their research, the authors also describe another process called Terrain Aided Navigation System (TANS), which is very similar. The difference is that, with TANS, the terrain has been previously mapped and stored in a database accessible to the INS. This reduces

uncertainty in the measurements and increases performance, but relies on a priori information. The research herein relies on a similar SLAM-like algorithm to generate landmarks. While the image processing side of the SLAM algorithm is not the focus of this thesis, locating and tracking landmarks is a large part. These tracked landmark locations become part of the overall state vector in the Kalman filter in order to aid the INS. In essence, the SLAM algorithm allows the aircraft to develop its own "a priori" information for landmarks it finds along the way. The University of Sydney study gives credibility to a pivotal concept on which this thesis depends. This research will show that the environments a military aircraft will encounter pose significant challenges not addressed in the work described above, and solutions to these challenges will be posed.

1.4.4 Image Aiding an EGI during GPS Drop Outs. The NAVSYS Corporation has recently been researching the use of image-based updates to assist EGI systems during periods of GPS denial [4][23]. The principles applied are very similar to those used herein: two-dimensional landmarks are tracked to update the inertial navigation solution. Three-dimensional object models were generated of suitable landmarks in the experimental flight area. These models were localized through multiple images and a target geo-location algorithm while GPS was available to the EGI [23]. The model database included a three-dimensional position for each landmark and distinct image attributes. This data bank of landmarks served to update the EGI when GPS was later denied during the flight test. The stored image attributes helped to find and identify the landmarks, while the three-dimensional position provided a sort of a priori information. Update rates of 1-2 per minute proved sufficient for short periods of GPS denial. The

experimental results showed great promise for image aiding INS in the manner proposed herein.

The NAVSYS work, although very similar, differs quite significantly from this thesis. Firstly, NAVSYS relied heavily on a priori information gathered while GPS was available. This research will attempt to rely on landmarks of opportunity (LOO) that have not been previously surveyed (no a priori information). Additionally, the NAVSYS process requires that the surveyed landmarks be revisited during the GPS outages. The aircraft also maintained the same altitude during the entire flight. These assumptions are not robust enough for dynamic profiles that military applications may undergo. Lastly, the periods of GPS denial were relatively short, while this thesis will explore the performance of image aiding during the course of an entire flight profile. Although different in many important ways, the NAVSYS study validates to the use of two-dimensional images to aid INS.

1.4.5 Stochastic Image Correspondence. In order to have any benefit, the location of landmarks, within successive images, must be a function of the inertial state of the aircraft taking the pictures. Relating landmarks in different images is termed the correspondence problem, and can be simplified by using inertial navigation information. Combining conventional epipolar correspondence geometry with stochastic projection modeling, Veth, Raquet, and Pachter were able to decrease the uncertainty projection of landmarks significantly in subsequent image frames [28]. The reduced uncertainty decreases the processing cost of feature tracking algorithms that are critical to image-aided INS. This research is important because it characterizes, stochastically, the image

correspondence problem upon which image aiding rests. Because the landmark projection is modeled stochastically, and the INS is updated via an extended Kalman filter, superior performance can be expected over any other method [14][15]. This stochastic model of the landmark projection will be incorporated into the research herein.

1.4.6 Experimental Validation – Peeping Talon. Giebner and Raquet performed an image aided INS flight test at the USAF TPS in 2002, nicknamed "*Peeping Talon*" [6]. A USAF T-38 was equipped with a Honeywell H-764G Embedded GPS/INS (EGI), two video cameras, and two Ashtech Z-surveyor semi-codeless receivers to provide truth data. The experiment collected inertial measurements from the H-764G EGI and imagery from the two cameras as it orbited over a predetermined target area. A small number of features were chosen to be landmarks, and manually tracked through the video frames (feature tracking algorithms were not employed). The inertial measurements and the manual video measurements were fed into a feed-forward extended Kalman filter, and compared to the truth data obtained by the Ashtech receivers.

The experimental results showed that image aiding could deliver close to GPS quality performance. This research validates the effectiveness of image aiding aboard USAF aircraft carrying USAF INS, but leaves much to expand upon. First of all, feature tracking was accomplished manually. This induces some man-made errors and reduces some machine-made errors inherent to feature tracking algorithms. Second, the inertial measurements were taken from the relatively high quality H-764G. Third, the flight path varied little, and revisited the same set of landmarks over and over. Fourth, the camera used looked out the side of the aircraft, where this test will look along the longitudinal

axis. Lastly, only a small number of landmarks were used, and the geometry of these landmarks varied little over the flight.

The research herein will expand upon the Peeping Talon work, in order to provide a more robust solution to the fore-mentioned military applications. First, an online feature tracker will be implemented and modeled. Many more features will be automatically identified and simultaneously tracked. The inertial measurements will be taken from free-running MEMS and low-quality INS units. Last, experimental flight trajectories will model appropriate UAV and weapons profiles, and not allow for the artificial revisiting of landmarks. The thrust of this thesis will be to take concepts proven in research like Peeping Talon, and develop it for practical applications.

1.4.7 Genesis of this Research. This research effort is an extension of work accomplished as an AFIT PhD dissertation in 2005 [26]. Veth successfully developed an end-to-end optical-inertial navigation system that employed binocular vision. Borrowing the Scale Invariant Feature Transform (SIFT© [11]) to accomplish the image processing, Veth employed an extended Kalman filter to accomplish the INS aiding. He solved the computational burden of feature matching and correspondence by constraining the search with estimates derived by the INS [28]. The system was built around a MEMS-grade inertial measurement unit (IMU) and two digital still cameras. The use of binocular vision enabled 3-dimensional resolution from 2-dimensional images. Veth accomplished meter level position precision in indoor environments, and flight test data showed similar promise, but lacked a faithful truth source. The system developed by Veth is very robust

and functional, and it truly incorporates zero a priori navigation with no artificialities that skew analysis.

The system used in this research incorporates the same MEMS IMU, feature generation algorithm, cameras and underlying algorithmic architecture used by Veth, but concentrates on the monocular case. Additionally, the analysis in this research benefits from precision truth data, which was not available to Veth at the time. Veth's work proved that a practical SLAM-based algorithm can in fact provide precision navigation in dynamic environments, validating the research herein. Building upon his work allows the extension to the military aviation environment. This environment exhibits challenges not encountered in the indoor case or at low altitude environments discussed to this point.

1.5 Methodology

This thesis begins with the mathematical background needed to understand inertial navigation, Kalman filtering and basic optics principles. Inertial navigation theory relies on kinematics and the basic laws of physics, which are described in detail in Chapter 2. A precise earth model is developed, as well as the characterization of inertial sensors. The INS model required for aiding, and how it relates to Kalman filtering, are described.

Chapter 3 develops the equations that relate the optical measurements to the INS states. These equations are then linearized for use in the extended Kalman filter. A rigorous study of the observability of these measurement equations is accomplished in an effort to design an optimal algorithm. The nature of surveyed landmarks and landmarks

of opportunity is explored. A set of principles for optical aiding design is developed, leading to the final algorithm.

Chapter 4 describes how simulation was used to validate and generate the flight test algorithm and apparatus. The flight test program is described from a technical design and calibration standpoint. The flight test data points were chosen to isolate and evaluate the factors influencing performance (derived in Chapter 3). Characterization of the performance is accomplished through statistical rigor. Characterizations will be made by examining the statistical performance over many data events, and not on single *best case* data runs.

Finally, flight test data will be used to form conclusions about the nature of optical-inertial navigation. The system under test (SUT) will be evaluated and recommendations for improvements made. Future areas of study will be also recommended, based on flight test shortcomings or predicted areas of fruitful study.

II. Mathematical Background

This chapter describes the mathematics and physics required to understand inertial navigation and the associated aiding process. It describes the notation and models to be used throughout this research, as well as the physical dynamics driving the process. Error analysis tools will also be described.

This chapter establishes the required notation, frames of reference, basic Earth model, inertial sensor models, and Inertial Navigation System (INS) mechanization. A development of linearized systems and extended Kalman filters (EKF) follows, enabling INS aiding. Lastly, the optics model is described. The bulk of this discussion is derived from the work of Tritteron and Weston on inertial navigation [23], Maybeck on Kalman filtering [14][15], and Veth on inertial optical fusion [26].

2.1 Mathematical Notation

The notation used herein is described below:

- **Scalars:** upper or lower case italic letters, (e.g., a or A).
- **Vectors:** lower case bold letters, (e.g., $\mathbf{x}$). All vectors are column vectors; x_i is the i^{th} scalar component of $\mathbf{x}$ in the i^{th} row of $\mathbf{x}$.
- **Unit Vectors:** denoted with the *check* character (e.g., $\check{\mathbf{x}}$). Unit Vectors have unit length defined by the two norm (e.g., $\|\check{\mathbf{x}}\| = 1$).

- **Matrices:** upper case letters in bold font. (e.g., $\mathbf{X}$) X_{ij} is the scalar component of $\mathbf{X}$ in the i^{th} row and j^{th} column.
- **Transpose:** a vector or matrix transpose is indicated by a superscript $\mathbf{T}$ (e.g., $\mathbf{x}^{\mathbf{T}}$ or $\mathbf{X}^{\mathbf{T}}$).
- **Estimated variables:** variables which are estimates of random variables are denoted by the *hat* character (e.g., $\hat{x}$).
- **Computed Variables:** variables which are computed and therefore corrupted by errors are denoted by the *tilde* character (e.g., $\tilde{x}$).
- **Measured Variables:** variables which are measured and therefore corrupted by errors are denoted by the *bar* character (e.g., $\bar{x}$).
- **Homogeneous Coordinates:** vectors in homogeneous coordinate form are denoted by an *underline* (e.g., $\underline{\mathbf{x}}$). Homogeneous coordinate vectors have a value of 1 in the last element.
- **Direction Cosine Matrices (DCM):** a DCM that transforms from frame *a* to frame *b* is denoted by $\mathbf{C}_a^b$.
- **Reference Frame:** a vector expressed in a particular reference frame is denoted by a superscript letter referring to said frame (e.g., $\mathbf{x}^a$ is expressed in the *a-frame*)
- **Relative Position or Motion:** positions and motions relative to two different frames are denoted by two subscript letters corresponding to the two frames of interest. (e.g., $\boldsymbol{\omega}_{ab}$ represents the rotation rate vector of *b relative to a*.)

2.2 Reference Frames

Inertial navigation and aiding requires an understanding of the reference frames used to express 3-dimensional vectors and measurements. The reference frames used herein are three-dimensional Cartesian frames with an orthonormal basis in $\Re^3$, described below:

2.2.1 True Inertial Frame (I-frame). Theoretical frame where Newton's laws apply. It is non-accelerating and non-rotating, but has no defined origin (due to relativity).

2.2.2 Earth-Centered Inertial Frame (i-frame). This frame has an origin at the center of the Earth. The z-axis points to the North Star. The x-axis falls on the equatorial plane, and points to the first star of Aires. The y-axis is orthogonal to both and falls on the equatorial plane. This frame is non-rotating, but does move with the Earth as the Earth rotates about the sun (and the sun about the galaxy). For the purposes of navigation on the Earth, the *i-frame* can be approximated as a true inertial frame (due to the slow rate of the Earth about the sun, etc).

2.2.3 Earth-Centered Earth-Fixed Frame (e-frame or ECEF frame). Like the *i-frame*, the *e-frame* has its origin at the center of the Earth, but it *does* rotate with the Earth; it is fixed to the planet. The z-axis aligns with the North pole. The x-axis falls on the equatorial plane, and points to the Greenwich meridian. The y-axis again falls on the equatorial plane, and points to 90 degrees east longitude.

2.2.4 Navigation Frame (n-frame). This is a local geographic frame, with its origin at the location of the navigation system. The x and y-axis point to North and East,

respectively. The z-axis points down. Down is described by the direction of the gravity vector for a given location on the Earth. This North-East-Down (NED) convention will be used in other frames as well. See Figure 2.1 for illustrations of the *i.e. and n-frame.*

2.2.5 Body Frame (b-frame). This frame is rigidly attached to the vehicle. The origin of the *b-frame* is co-located with the *n-frame*, but the axes rotate with the aircraft. The x, y and z-axis point out the nose, right wing, and bottom of the aircraft, respectively. See Figure 2.2.

2.2.6 Camera Frame (c-frame). This frame is rigidly attached to the camera. The origin is at the camera's optical center. The x and y-axis point up and right, respectively. The z-axis points out the camera lens, perpendicular to the focal plane. It is useful to say that the x and y-axis align to the camera's detector array (discussed later). See Figure 2.3.

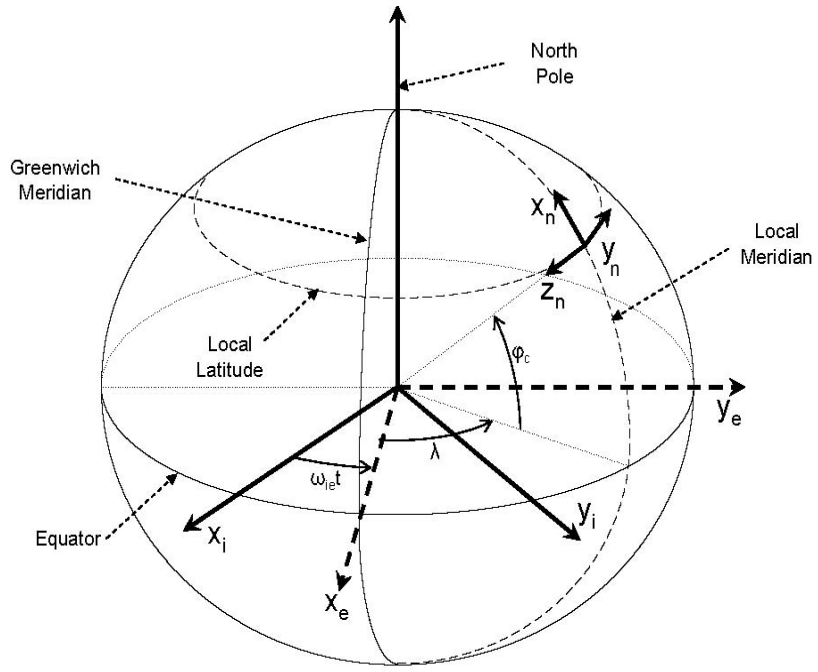

Figure 2.1: Earth-Centered Inertial, Earth-Centered Earth-Fixed, and Navigation Frame. The Earth Fixed and Earth-Centered Earth-Fixed frames originate at the Earth's center of mass, while the navigation frame is fixed to a vehicle [26].

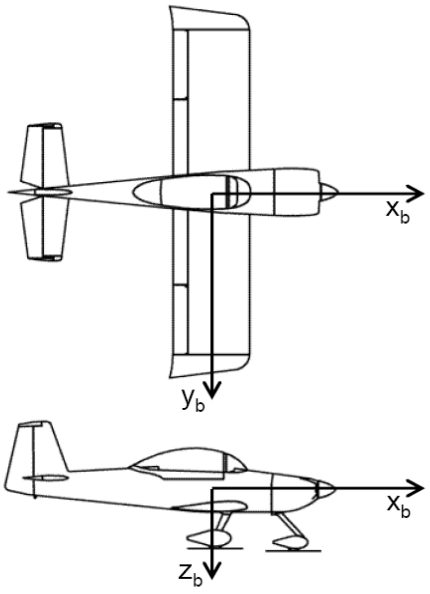

Figure 2.2: Aircraft body frame illustration. The origin of the *b-frame* is co-located with the *n-frame*, but the axes rotate with the aircraft [26].

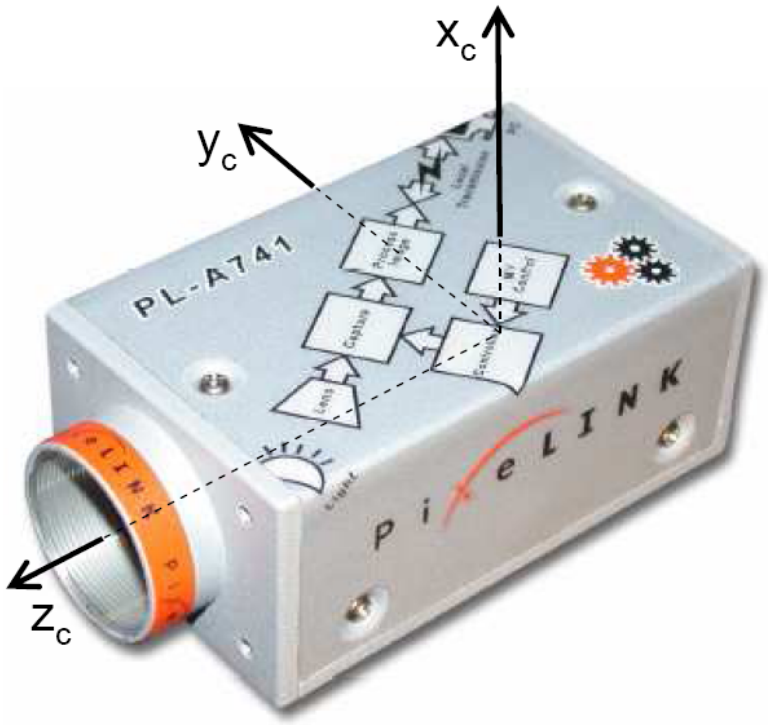

Figure 2.3: Camera frame illustration. The camera reference frame originates at the center of the focal plane [26].

An understanding of reference frames is important to express vectors properly. The transformation between reference frames can now be addressed.

2.3 Transforms Between Reference Frames

Any vector can be expressed (or resolved) in any of the frames described. If a vector is resolved in one frame, it will have the superscript corresponding to that frame. See Figure 2.4 below.

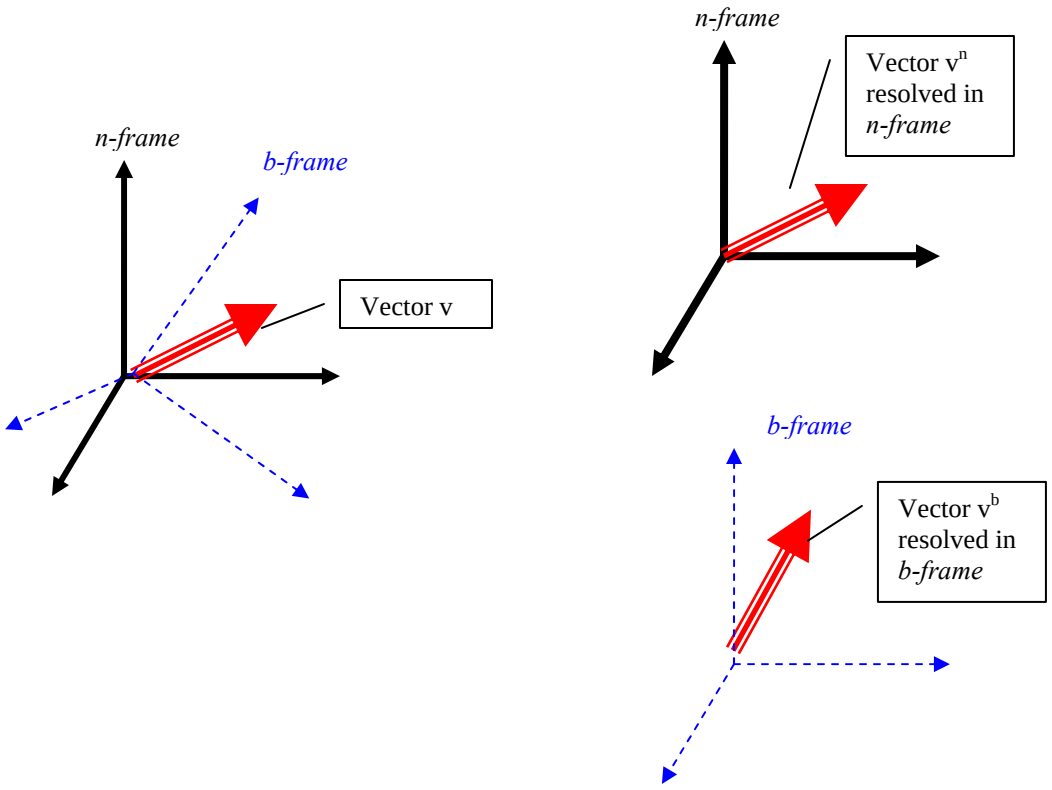

Figure 2.4: Expressing vectors in different frames. The same vector can take on different expressions when resolved in different frames. The magnitude is unaffected.

To express a vector in one frame, when given a vector resolved in another, a Direction Cosine Matrix (DCM) can be used to perform a coordinate transformation. The DCM can be generated from a series of Euler angles to perform an ordered set of single

axis rotations on the vector of interest. Euler angles are elements of a three-parameter vector corresponding to a specific sequence of single-axis rotations to transform from one reference frame to another. Commonly, Euler angles express the transformation from the navigation to body frame of an aircraft: yaw, pitch and roll. Performing the rotations in this order is known as the 3-2-1 convention [23]. This convention will be used herein.

Direction Cosine Matrices are four-parameter transformations expressed as a 3×3 matrix [23]. The matrix consists of the inner product (or Cosines) of each basis unit vector in one frame with each basis unit vector in another frame. The DCM is used to transform vectors resolved in one frame to another, as seen below:

$$\mathbf{v}^b = \mathbf{C}_a^b \mathbf{v}^a \tag{2.1}$$

When used to transform right-hand Cartesian coordinates, the DCM has the following properties:

$$\begin{aligned} &Det(\mathbf{C}_a^b) \equiv \left|\mathbf{C}_a^b\right| = 1 \\ &\mathbf{C}_b^a = \left(\mathbf{C}_a^b\right)^{-1} = \left(\mathbf{C}_a^b\right)^{\mathsf{T}} \\ &\mathbf{C}_a^c = \mathbf{C}_b^c \mathbf{C}_a^b \end{aligned} \tag{2.2}$$

When a DCM changes over time, due to the frames of interest rotating with respect to each other, the DCM satisfies the following differential equation [23]:

$$\dot{\mathbf{C}}_a^b = \mathbf{C}_a^b \Omega_{ba}^a \tag{2.3}$$

where Ω_{ba}^a is the skew symmetric form of the rotation vector from frame b to a, resolved in a ($\boldsymbol{\omega}_{ba}^a$). The skew symmetric operator of a vector is defined by $\boldsymbol{\omega}\times$:

$$\boldsymbol{\omega} = \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} \qquad \boldsymbol{\omega}\times = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} \tag{2.4}$$

Direction Cosine Matrices are essential to apply when dealing with vectors resolved in different frames. Vectors from different frames cannot be used together, until they are transformed into the same reference frame. With an understanding of how vectors are expressed, the process of modeling physical systems can now be addressed.

2.4 Physical System Modeling

Physical systems and their associated dynamics equations can often be written and arranged into convenient matrix form. This allows the use of linear algebra to accomplish much of the modeling needed in INS applications. The dynamics herein are modeled as systems of linear differential equations.

2.4.1 Linear Systems. Many physical systems can be modeled such that they fit the format of a linear system of equations [14]; shown below.

$$\dot{\mathbf{x}}(t) = \mathbf{F}(t)\mathbf{x}(t) + \mathbf{B}(t)\mathbf{u}(t)\,; \quad \mathbf{x}(t_0) = \mathbf{x}_0 \tag{2.5}$$

In the linear time-varying case, $\mathbf{F}(t)$ and $\mathbf{B}(t)$ are time-varying matrices. For a given set of initial conditions ($\mathbf{x}_0$) and input ($\mathbf{u}(t)$), Equation (2.5) has a unique solution trajectory, $\mathbf{x}(t)$ [14]; given by:

$$\mathbf{x}(t) = \Phi(t,t_0)\mathbf{x}_0 + \int_{t_0}^{t} \Phi(t,\tau)\mathbf{B}(\tau)\mathbf{u}(\tau)d\tau \tag{2.6}$$

The state transition matrix $\Phi(t,t_0)$ takes the state $\mathbf{x}(t)$ from time t_0 to time t for the homogeneous case (Equation (2.6) without the last term on the right hand side), and is defined by:

$$\frac{d[\Phi(t,t_0)]}{dt} = \mathbf{F}(t)\Phi(t,t_0)\,; \quad \Phi(t_0,t_0) = \mathrm{I} \tag{2.7}$$

Significant properties of $\Phi(t,t_0)$ include:

$$\Phi(t_3,t_1) = \Phi(t_3,t_2)\Phi(t_2,t_1) \tag{2.8}$$

$$\Phi(t_1,t_2)\Phi(t_2,t_1) = \Phi(t_1,t_1) = \mathrm{I} \tag{2.9}$$

$$\Phi(t_1,t_2)^{-1} = \Phi(t_2,t_1) \tag{2.10}$$

In the linear time-invariant case, where $\mathbf{F}(t) = \mathbf{F}$, $t \in \Re$

$$\frac{d[\Phi(t\text{-}t_0)]}{dt} = \mathbf{F}\Phi(t\text{-}t_0); \quad \Phi(0) = \mathrm{I} \tag{2.11}$$

$$\Phi(t,t_0) = \Phi(t\text{-}t_0) = \mathbf{e}^{\mathbf{F}(t\text{-}t_0)} \tag{2.12}$$

2.4.2 Non-Linear Systems. Most physical systems do not allow for linear models. In the non-linear case, the physical system can often be expressed as the nonlinear differential equation [15]; shown below.

$$\dot{\mathbf{x}}(t) = \mathbf{f}[\mathbf{x}(t),\mathbf{u}(t),t]; \quad \mathbf{x}(t_0) = \mathbf{x}_0 \tag{2.13}$$

where $\mathbf{f}[\mathbf{x}(t),\mathbf{u}(t),t]$ is a vector of functions with arguments including the state vector ($\mathbf{x}(t)$), the input vector ($\mathbf{u}(t)$), and time (t). The non-trivial solution $\mathbf{x}(t)$ is the result of a nonlinear differential equation.

Linear properties may be retained by making linear approximations of the nonlinear solution, about nominal trajectory values. For small changes about a nominal trajectory, many systems can be well modeled to first order through the use of the Taylor Series expansion [15] shown below.

$$[\dot{\mathbf{x}}(t) - \dot{\mathbf{x}}_{\mathbf{n}}(t)] = \left.\frac{\partial \mathbf{f}[\mathbf{x}, \mathbf{u}(t), t]}{\partial \mathbf{x}}\right|_{\mathbf{x}=\mathbf{x}_{\mathbf{n}}(\mathbf{t})} [\mathbf{x}(t) - \mathbf{x}_{\mathbf{n}}(t)] + H.O.T. \tag{2.14}$$

where $\mathbf{x}_{\mathbf{n}}(t)$ is the nominal trajectory about which the linearization occurs. Note that $\mathbf{u}(t)$ is assumed to be unperturbed ($\mathbf{u}(t) - \mathbf{u}_{\mathbf{n}}(t) = \underline{\mathbf{0}}$), thus eliminating the partial with respect to $\mathbf{u}(t)$. If the system is well modeled to first order, the higher order terms (H.O.T.) may be neglected [15]. When $\mathbf{x}(t)$ is redefined to be a perturbation about the nominal,

$$\mathbf{x}(t) = \mathbf{x}_{\mathbf{n}}(t) + \delta\mathbf{x}(t) \tag{2.15}$$

Equation (2.14) is approximated to first order as:

$$\delta\dot{\mathbf{x}}(t) = \left.\frac{\partial \mathbf{f}[\mathbf{x}, \mathbf{u}(t), t]}{\partial \mathbf{x}}\right|_{\mathbf{x}=\mathbf{x}_{\mathbf{n}}(\mathbf{t})} \delta\mathbf{x}(t) \tag{2.16}$$

or

$$\delta\dot{\mathbf{x}}(t) = \mathbf{F}[t; \mathbf{x}_{\mathbf{n}}(t)]\delta\mathbf{x}(t) \tag{2.17}$$

where $\mathbf{F}[t; \mathbf{x}_{\mathbf{n}}(t)]$ is the matrix of the partial derivatives of $\mathbf{f}[\mathbf{x}, \mathbf{u}(t), t]$ [15]:

$$\mathbf{F}[t;\mathbf{u}(t),\mathbf{x_n}(t)] = \left.\frac{\partial \mathbf{f}[\mathbf{x},\mathbf{u}(t),t]}{\partial \mathbf{x}}\right|_{\mathbf{x}=\mathbf{x_n}(t)} = \left.\begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_1} & \cdots & \frac{\partial f_n}{\partial x_n} \end{bmatrix}\right|_{\mathbf{x}=\mathbf{x_n}(t)} \tag{2.18}$$

In error analysis, the estimated value of the state vector ($\hat{\mathbf{x}}(t)$) is equal to the true value of the state vector ($\mathbf{x}(t)$) plus some error ($\delta\mathbf{x}(t)$).

$$\hat{\mathbf{x}}(t) = \mathbf{x}(t) + \delta\mathbf{x}(t) \tag{2.19}$$

If Equation (2.19) is substituted for Equation (2.15), Equation (2.17) becomes a nonlinear differential equation for the error ($\delta\mathbf{x}(t)$) in the state estimate ($\hat{\mathbf{x}}(t)$), linearized about the true state ($\mathbf{x}(t)$).

In the case where $\mathbf{u}(t)$ is not known (i.e. $\mathbf{u}(t) - \mathbf{u_n}(t) \neq \underline{\mathbf{0}}$), the error ($\delta\mathbf{u}(t)$) can be modeled as well by linearizing about a nominal input trajectory. However, this compromises the validity of the small perturbation approximation, used in Equation (2.16) [15]. If $\delta\mathbf{u}(t)$ cannot be approximated to $\mathbf{0}$ for all time, then it can be modeled as an error state according to Equations (2.20)(2.21)(2.22).

$$\delta\dot{\mathbf{x}}(t) = \mathbf{F}[t;\mathbf{x_n}(t),\mathbf{u_n}(t)]\delta\mathbf{x}(t) + \mathbf{B}[t;\mathbf{x_n}(t),\mathbf{u_n}(t)]\delta\mathbf{u}(t) \tag{2.20}$$

$$\mathbf{F}[t;\mathbf{x_n}(t),\mathbf{u_n}(t)] = \left.\frac{\partial \mathbf{f}[\mathbf{x},\mathbf{u},t]}{\partial \mathbf{x}}\right|_{\mathbf{x}=\mathbf{x_n}(t),\mathbf{u}=\mathbf{u_n}(t),t} = \left.\begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_1} & \cdots & \frac{\partial f_n}{\partial x_n} \end{bmatrix}\right|_{\mathbf{x}=\mathbf{x_n}(t),\mathbf{u}=\mathbf{u_n}(t),t} \tag{2.21}$$

$$\mathbf{B}[t;\mathbf{x_n}(t),\mathbf{u_n}(t)] = \left.\frac{\partial \mathbf{f}[\mathbf{x},\mathbf{u},t]}{\partial \mathbf{u}}\right|_{\mathbf{x}=\mathbf{x_n}(t),\mathbf{u}=\mathbf{u_n}(t),t} = \left.\begin{bmatrix} \frac{\partial f_1}{\partial u_1} & \cdots & \frac{\partial f_1}{\partial u_m} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial u_1} & \cdots & \frac{\partial f_n}{\partial u_m} \end{bmatrix}\right|_{\mathbf{x}=\mathbf{x_n}(t),\mathbf{u}=\mathbf{u_n}(t),t} \tag{2.22}$$

With the higher order derivatives neglected, $\delta\mathbf{x}(t)$ becomes the solution to the time-varying, linear differential Equation (2.20). Given a piecewise-continuous $\mathbf{F}[t;\mathbf{x_n}(t),\mathbf{u_n}(t)]$ and $\mathbf{B}[t;\mathbf{x_n}(t),\mathbf{u_n}(t)]\partial\mathbf{u}(t)$, $\delta\mathbf{x}(t)$ has a unique solution [15]:

$$\delta\mathbf{x}(t) = \Phi(t,t_0)\delta\mathbf{x}_0 + \int_{t_0}^{t} \Phi(t,\tau)\mathbf{B}(\tau)\delta\mathbf{u}(\tau)d\tau \tag{2.23}$$

where $\mathbf{B}(\tau)$ is defined by Equation (2.22), and $\Phi(t,t_0)$ is defined by Equation (2.7) using $\mathbf{F}(t)$ found in Equation (2.21).

These non-linear perturbation models will be used when implementing the extended Kalman filter, discussed later. The extended Kalman filter will be used to aid the Inertial Navigation System, which will now be discussed.

2.5 Inertial Navigation

Inertial navigation is accomplished by integrating specific force and angular rate measurements to produce an estimate of the navigation state: position, velocity and attitude. Only an estimate can be attained due to imperfections in sensors and approximations of the true system dynamics. This section will cover the dynamics, mechanization, and error propagation of an INS.

2.5.1 Shape of the Earth. In order to navigate on the Earth, one must know its shape. There are many models for the Earth, but the World Geodetic System of 1984 (WGS84) model will be used for this research [23]. WGS84 models the Earth as an ellipsoid, wider along the equator. The mean radius of the Earth (R_0), the meridian radius of curvature (R_N), and the transverse radius of curvature (R_E) are modeled as:

$$R_0 = \sqrt{R_N R_E} \tag{2.24}$$

$$R_N = \frac{R(1-e^2)}{(1-e^2 \sin^2 L)^{3/2}} \tag{2.25}$$

$$R_E = \frac{R}{(1-e^2 \sin^2 L)^{1/2}} \tag{2.26}$$

These radii are functions of latitude (L), eccentricity of the ellipsoid model (e), and the semi-major axis (R); defined by WGS84:

$$R = 6378137.0 \text{ meters} \tag{2.27}$$

$$e = 0.0818191908426 \tag{2.28}$$

2.5.2 Gravitational Model. Modeling gravity is critical to successful inertial navigation. The subtle changes in the gravitational field must be taken into account. The most obvious factor affecting the strength of the gravitational field is the distance from the center of the Earth. For this research, gravity will be modeled as follows [23]:

$$g(h) = \frac{g(0)}{\left(1 + h / R_o\right)^2} \tag{2.29}$$

where h is the height above the WGS-84 ellipsoid (Earth model), R_o is the Earth's radius defined above, and $g(0)$ is the magnitude of gravity at the ellipsoid surface:

$$g(0) = 9.780318(1 + 5.3024 \times 10^{-3} \sin^2 L - 5.9 \times 10^{-6} \sin^2 2L) \quad m/s^2 \tag{2.30}$$

Centripetal acceleration due to the rotation of the Earth affects the magnitude and direction of the effective gravitational field at every point on the Earth. Figure 2.5 illustrates this effect. Local gravity ($\mathbf{g}_l$) becomes a function of the actual gravitational field vector ($\mathbf{g}$), the Earth's rotation vector (ω_{ie}) and the specific position on the Earth ($\mathbf{p}$), and is defined:

$$\mathbf{g}_l = \mathbf{g} - \boldsymbol{\omega}_{ie} \times [\boldsymbol{\omega}_{ie} \times \mathbf{p}] \tag{2.31}$$

where $\|\mathbf{g}\|$ is defined by Equation (2.29). The *n-frame* is defined tangent to the ellipsoid. The true gravitational field varies slightly, but the difference between down defined by the true gravitational field and the *n-frame* is small. For the purposes of this discussion assume that the gravitational field points down in the *n-frame*:

$$\mathbf{g} \equiv \begin{bmatrix} 0 & 0 & g(h) \end{bmatrix}^{\mathsf{T}} \tag{2.32}$$

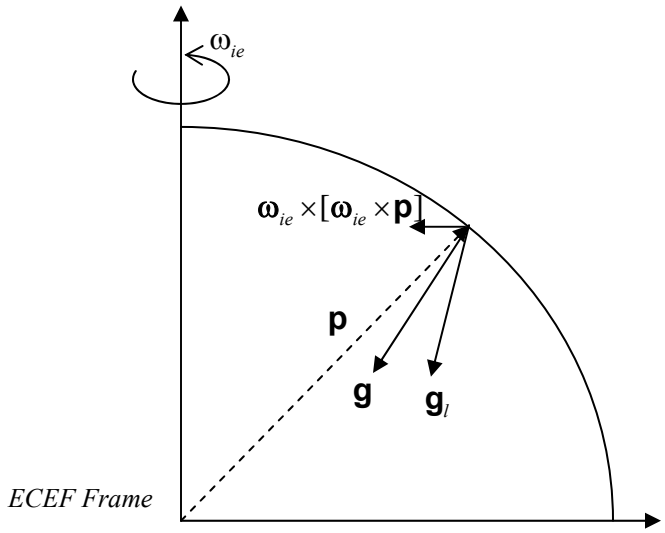

Figure 2.5: Effects of centripetal force on the local gravitational field.

For terrestrial navigation, the Earth's rotation (sidereal angular rate) can be modeled as constant [23]:

$$\boldsymbol{\omega}_{ie}^{e} = \begin{bmatrix} 0 & 0 & \Omega_{ie} \end{bmatrix}^{\mathsf{T}} \tag{2.33}$$

However, this rotation vector, resolved in the *n-frame,* is a function of latitude:

$$\boldsymbol{\omega}_{ie}^{n} = \begin{bmatrix} \Omega_{ie}\cos L & 0 & -\Omega_{ie}\sin L \end{bmatrix}^{\mathsf{T}} \tag{2.34}$$

where Ω_{ie} and L are the scalar Earth sidereal angular rate and latitude respectively. WGS84 defines:

$$\Omega_{ie} = 7.292115\times10^{-5}\, rad/s \tag{2.35}$$

Combining Equations (2.31),(2.4) and (2.34) yields a final expression for local gravity, resolved in the *n-frame*:

$$\mathbf{g}_{l}^{n} = \mathbf{g}^{n} - \frac{\Omega_{ie}{}^{2}(R_0+h)}{2}\begin{bmatrix} sin2L \\ 0 \\ 1+cos2L \end{bmatrix} \tag{2.36}$$

2.5.3 Accelerometer Sensor Model. INS systems typically incorporate a triad of accelerometers, arranged at the origin, pointing orthogonally along each of the three *b-frame* axes. Despite the name, accelerometers do not measure acceleration directly. Accelerometers measure specific force. Specific force is the sum of the acceleration experienced due to gravity, and the inertial acceleration experienced by the body [23]:

$$\mathbf{f}(t) = \left.\frac{d^{2}\mathbf{p}(t)}{dt^{2}}\right|_{i} - \mathbf{g}_{l} \tag{2.37}$$

where $\mathbf{f}(t)$ is the three-dimensional vector of specific forces measured by the accelerometers. The position of the *b-frame* $\mathbf{p}(t)$ is resolved in inertial space, and $\mathbf{g}_l$ is the local gravitational vector at the specific location $\mathbf{p}(t)$.

In order to be used by an INS, these accelerometers must be modeled and their errors characterized. The simplified accelerometer model used herein has the specific force measurement corrupted by a bias and an additive white Gaussian noise [26].

$$\bar{\mathbf{f}}^b = \mathbf{f}^b + \mathbf{a}^b + \mathbf{w}_a^b \tag{2.38}$$

where $\mathbf{a}^b$ is the accelerometer bias, $\mathbf{w}_a^b$ is the additive white Gaussian noise, and $\bar{\mathbf{f}}^b$ is the corrupted specific force measurement, resolved in the *b-frame*.

The accelerometer bias will be modeled as a first order Gauss-Markov process, defined below:

$$\dot{\mathbf{a}}^b = -\frac{1}{\tau_a}\mathbf{a}^b + \mathbf{w}_{a_{bias}}^b \tag{2.39}$$

where τ_a is the time constant and $\mathbf{w}_{a_{bias}}^b$ is the process driving white noise.

2.5.4 Gyroscope Sensor Model. INS systems typically incorporate a triad of angular rate gyros, aligned with each of the three *b-frame* axes. These rate gyros measure the angular rotation rate of the body frame relative to inertial space, and are represented by $\boldsymbol{\omega}_{ib}(t)$ [23].

When dealing with the *b-frame* relative to the *n-frame*, the Euler angles represent roll, pitch and yaw respectively. When put into vector form, and resolved in the *n-frame*, the result is the attitude state vector [23]:

$$\Psi_{nb}^{n} = \begin{bmatrix} \phi & \theta & \psi \end{bmatrix}^{\mathsf{T}} \tag{2.40}$$

The rate gyros will be modeled in a similar manner to the accelerometers. The simplified rate gyro model used herein has the angular rate measurement corrupted by a bias and an additive white Gaussian noise [26].

$$\bar{\boldsymbol{\omega}}_{ib}^{b} = \boldsymbol{\omega}_{ib}^{b} + \mathbf{b}^{b} + \mathbf{w}_{b}^{b} \tag{2.41}$$

where $\mathbf{b}^{b}$ is the rate gyro bias, $\mathbf{w}_{b}^{b}$ is the additive white Gaussian noise, and $\bar{\omega}_{ib}^{b}$ is the corrupted angular rate measurement, resolved in the *b-frame*.

The rate gyro bias will also be modeled as a first order Gauss-Markov process:

$$\dot{\mathbf{b}}^{b} = -\frac{1}{\tau_{b}}\mathbf{b}^{b} + \mathbf{w}_{b_{bias}}^{b} \tag{2.42}$$

where τ_{b} is the time constant and $\mathbf{w}_{b_{bias}}^{b}$ is the process driving white noise [26].

2.5.5 Strapdown INS mechanization. The system dynamics driving inertial navigation will not be covered in depth here. A full derivation of the system dynamics

can be found in [23] and [26]. This section will cover the mechanization equations used to estimate the navigation state (based on the inertial measurements), but will not explain their derivation.

2.5.5.1 Attitude Mechanization. Recall Equation (2.40). The Euler angles describe the relative orientation of two frames. The Euler angles of concern to an INS are those relating the body frame to the navigation frame of choice. This research will focus on the *n-frame* for navigation, although the use of the *i-frame* and *e-frame* are perfectly valid choices. One dimensional rotations about each of the axis are defined by [23]:

$$\text{Rotation } \psi \text{ about the z-axis, } C_1 = \begin{bmatrix} \cos\psi & \sin\psi & 0 \\ -\sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.43}$$

$$\text{Rotation } \theta \text{ about the y-axis, } C_2 = \begin{bmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{bmatrix} \tag{2.44}$$

$$\text{Rotation } \phi \text{ about the x-axis, } C_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi & \cos\phi \end{bmatrix} \tag{2.45}$$

The order in which the individual Euler angle rotations are applied does effect the outcome. Applying the 3-2-1 convention to the *n-frame* to *b-frame* DCM yields:

$$\mathbf{C}_n^b = \mathbf{C}_3\mathbf{C}_2\mathbf{C}_1 \tag{2.46}$$

$$\mathbf{C}_b^n = \left(\mathbf{C}_n^b\right)^{\mathsf{T}} = \mathbf{C}_1^{\mathsf{T}}\mathbf{C}_2^{\mathsf{T}}\mathbf{C}_3^{\mathsf{T}} \tag{2.47}$$

$$\mathbf{C}_b^n = \begin{bmatrix} \cos\theta\cos\psi & -\cos\phi\sin\psi + \sin\phi\sin\theta\cos\psi & \sin\phi\sin\psi + \cos\phi\sin\theta\cos\psi \\ \cos\theta\sin\psi & \cos\phi\cos\psi + \sin\phi\sin\theta\sin\psi & -\sin\phi\cos\psi + \cos\phi\sin\theta\sin\psi \\ -\sin\theta & \sin\phi\cos\theta & \cos\phi\cos\theta \end{bmatrix} \tag{2.48}$$

The actual Euler angles are not used in the INS mechanization. Only the DCM $\mathbf{C}_b^n$ is needed. The Euler angles can be extracted from $\mathbf{C}_b^n$ in order to display attitude to the pilot, but this will be discussed later.

In [23], Titterton and Weston apply the differential Equation (2.3) to $\mathbf{C}_b^n$ and show that

$$\dot{\mathbf{C}}_b^n = \mathbf{C}_b^n\Omega_{nb}^b \tag{2.49}$$

where

$$\Omega_{nb}^b = \boldsymbol{\omega}_{nb}^b\times = \begin{bmatrix} 0 & -\dfrac{\partial\varepsilon_z}{\partial t} & \dfrac{\partial\varepsilon_y}{\partial t} \\ \dfrac{\partial\varepsilon_z}{\partial t} & 0 & -\dfrac{\partial\varepsilon_x}{\partial t} \\ -\dfrac{\partial\varepsilon_y}{\partial t} & \dfrac{\partial\varepsilon_x}{\partial t} & 0 \end{bmatrix} \tag{2.50}$$

where Ω_{nb}^{b} is the skew symmetric form of the *b-frame's* angular rate relative to the *n-frame*, resolved in the *b-frame* ($\boldsymbol{\omega}_{nb}^{b}$), and $\varepsilon_{x,y,z}$ are small angles about the 3 axes. For very small time steps (δt), the DCM can be propagated as follows:

$$\mathbf{C}_{b}^{n}(t+\delta t) = \mathbf{C}_{b}^{n}(t)[\mathrm{I}+\Omega_{nb}^{b}\delta t] \tag{2.51}$$

The measured angular rate experienced by the rate gyros ($\boldsymbol{\omega}_{ib}^{b}$) is the sum of the body-to-nav rate ($\boldsymbol{\omega}_{nb}^{b}$) and the nav-to-inertial rate resolved in the *b-frame* ($\mathbf{C}_{n}^{b}\,\boldsymbol{\omega}_{in}^{n}$), yielding:

$$\boldsymbol{\omega}_{nb}^{b} = \boldsymbol{\omega}_{ib}^{b} - \mathbf{C}_{n}^{b}\boldsymbol{\omega}_{in}^{n} \tag{2.52}$$

The angular rate of the *n-frame* relative to the *i-frame* ($\boldsymbol{\omega}_{in}^{n}$), is the sum of the sidereal angular rate ($\boldsymbol{\omega}_{ie}^{n}$), and the *transport rate* ($\boldsymbol{\omega}_{en}^{n}$), caused by the *n-frame* moving about the Earth [23]:

$$\boldsymbol{\omega}_{en}^{n} = \begin{bmatrix} \dfrac{v_E}{R_E+h} & -\dfrac{v_N}{R_N+h} & -\dfrac{v_E \tan L}{R_N+h} \end{bmatrix}^{\mathsf{T}} \tag{2.53}$$

$$\boldsymbol{\omega}_{in}^{n} = \boldsymbol{\omega}_{ie}^{n} + \boldsymbol{\omega}_{en}^{n} = \begin{bmatrix} \dfrac{v_E}{R_E+h} + \Omega_{ie}\cos L & -\dfrac{v_N}{R_N+h} & -\dfrac{v_E \tan L}{R_N+h} - \Omega_{ie}\sin L \end{bmatrix}^{\mathsf{T}} \tag{2.54}$$

where v_E and v_N are the east and north velocities of the *n-frame* respectively. R_E, R_N, h and L are the previously defined Earth radii of curvature, height above the Earth, and latitude of the *n-frame* respectively.

The measured *b-frame* to *n-frame* angular rate (resolved in the *b-frame*) becomes a function of the rate gyro measurements ($\bar{\boldsymbol{\omega}}_{ib}^{b}$), and the calculated sidereal rotation vector, dependant on latitude ($\tilde{\boldsymbol{\omega}}_{in}^{n}$, from Equation (2.54)) [23]:

$$\bar{\boldsymbol{\omega}}_{nb}^{b} = \bar{\boldsymbol{\omega}}_{ib}^{b} - \mathbf{C}_{n}^{b}\tilde{\boldsymbol{\omega}}_{in}^{n} \tag{2.55}$$

Combining Equations (2.49) and (2.55) yields the differential equation relating the DCM of interest ($\mathbf{C}_{b}^{n}$), with the measurements received by the rate gyros ($\bar{\boldsymbol{\omega}}_{ib}^{b} \Rightarrow \bar{\Omega}_{ib}^{b}$), and the calculated sidereal rate ($\tilde{\boldsymbol{\omega}}_{in}^{n} \Rightarrow \tilde{\Omega}_{in}^{n}$):

$$\dot{\mathbf{C}}_{b}^{n} = \mathbf{C}_{b}^{n}\bar{\Omega}_{ib}^{b} - \tilde{\Omega}_{in}^{n}\mathbf{C}_{b}^{n} \tag{2.56}$$

In order to be a valid DCM, the properties described in Section 2.3 must be maintained after each propagation cycle. Unfortunately, due to computational limitations, the computed DCM ($\tilde{\mathbf{C}}_{b}^{n}$) rarely has an orthonormal basis, nor is it unitary. This requires an ortho-normalization after each time step. Pachter develops an algorithm for this task [17]. Given an imperfect DCM resulting from an INS propagation process ($\mathbf{C}_{err}$):

$$\mathbf{C}_{err} = \begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{bmatrix} = \begin{bmatrix} \mathbf{w}_1 \\ \mathbf{w}_2 \\ \mathbf{w}_3 \end{bmatrix} \tag{2.57}$$

where $\mathbf{w}_n^{\mathsf{T}}$ is the n[th] row of $\mathbf{C}_{err}$. An orthogonal DCM can be created ($\mathbf{C}_{opt}$)

$$\mathbf{C}_{opt} = \mathbf{T}_{orth} \cdot \mathbf{C}_{err} = \begin{bmatrix} \frac{1}{2} \cdot \frac{1 + \mathbf{w}_1^{\mathsf{T}} \mathbf{w}_1}{\mathbf{w}_1^{\mathsf{T}} \mathbf{w}_1} & -\frac{\mathbf{w}_1^{\mathsf{T}} \mathbf{w}_2}{\mathbf{w}_1^{\mathsf{T}} \mathbf{w}_1 + \mathbf{w}_2^{\mathsf{T}} \mathbf{w}_2} & -\frac{\mathbf{w}_1^{\mathsf{T}} \mathbf{w}_3}{\mathbf{w}_1^{\mathsf{T}} \mathbf{w}_1 + \mathbf{w}_3^{\mathsf{T}} \mathbf{w}_3} \\ -\frac{\mathbf{w}_1^{\mathsf{T}} \mathbf{w}_2}{\mathbf{w}_1^{\mathsf{T}} \mathbf{w}_1 + \mathbf{w}_2^{\mathsf{T}} \mathbf{w}_2} & \frac{1}{2} \cdot \frac{1 + \mathbf{w}_2^{\mathsf{T}} \mathbf{w}_2}{\mathbf{w}_2^{\mathsf{T}} \mathbf{w}_2} & -\frac{\mathbf{w}_2^{\mathsf{T}} \mathbf{w}_3}{\mathbf{w}_2^{\mathsf{T}} \mathbf{w}_2 + \mathbf{w}_3^{\mathsf{T}} \mathbf{w}_3} \\ -\frac{\mathbf{w}_1^{\mathsf{T}} \mathbf{w}_3^{\mathsf{T}}}{\mathbf{w}_1^{\mathsf{T}} \mathbf{w}_1 + \mathbf{w}_3^{\mathsf{T}} \mathbf{w}_3} & -\frac{\mathbf{w}_2^{\mathsf{T}} \mathbf{w}_3^{\mathsf{T}}}{\mathbf{w}_2^{\mathsf{T}} \mathbf{w}_2 + \mathbf{w}_3^{\mathsf{T}} \mathbf{w}_3} & \frac{1}{2} \cdot \frac{1 + \mathbf{w}_3^{\mathsf{T}} \mathbf{w}_3}{\mathbf{w}_3^{\mathsf{T}} \mathbf{w}_3} \end{bmatrix} \mathbf{C}_{err} \tag{2.58}$$

For small time steps (δt), $\bar{\boldsymbol{\omega}}_{nb}^{b}$ (obtained from Equations (2.54) and (2.55)) can be applied to Equation (2.51) in skew symmetric form. The result is made orthogonal, producing the attitude DCM propagation equation (valid to first order). It is a function of the rate gyro measurements and the navigation state estimate.

$$\tilde{\mathbf{C}}_b^n(t+\delta t) = \mathbf{T}_{orth}\left[\hat{\mathbf{C}}_b^n(t) + \left(\bar{\boldsymbol{\omega}}_{ib}^{b} - \hat{\mathbf{C}}_n^b(t)\tilde{\boldsymbol{\omega}}_{in}^{n}\right)\times\right] \tag{2.59}$$

2.5.5.2 Position Mechanization. This research will concentrate on the *n-frame* resolution of the navigation state. Recall that the *n-frame* is attached to the body, and the axes are aligned to North, East, and Down (NED) respectively. For this reason, a

navigation position inside the *n-frame* is meaningless, as the body is always located at the origin. Instantaneous navigation velocities, however, can be expressed in the *n-frame* with meaning. For this reason, the navigation position states (latitude, longitude and height above the Earth), are external to the *n-frame*. They are obtained by integrating the *n-frame* velocities [23].

$$\dot{\mathbf{p}} = \begin{bmatrix} \dot{L} \\ \dot{l} \\ \dot{h} \end{bmatrix} = \mathbf{T}_{ne}\mathbf{v}_{en}^{n} = \begin{bmatrix} \frac{1}{R_N + h} & 0 & 0 \\ 0 & \frac{\sec L}{R_E + h} & 0 \\ 0 & 0 & -1 \end{bmatrix} \mathbf{v}_{en}^{n} = \begin{bmatrix} \frac{v_N}{R_N + h} \\ \frac{v_E \sec L}{R_E + h} \\ -v_D \end{bmatrix} \tag{2.60}$$

for

$$\mathbf{v}_{en}^{n} = \begin{bmatrix} v_N & v_E & v_D \end{bmatrix}^{\mathsf{T}} \tag{2.61}$$

and

$$\mathbf{T}_{ne} = \begin{bmatrix} \frac{1}{R_N + h} & 0 & 0 \\ 0 & \frac{\sec L}{R_E + h} & 0 \\ 0 & 0 & -1 \end{bmatrix} \tag{2.62}$$

where $\mathbf{p} = \begin{bmatrix} L & l & h \end{bmatrix}^T$ is the position vector (latitude, longitude and height above respectively), and $\mathbf{v}_{en}^{n}$ is the velocity of the *n-frame* relative to the *e-frame*, resolved in the *n-frame*. R_E and R_N are the previously defined Earth radii of curvature. The transformation $\mathbf{T}_{ne}$ relates the velocity ($\mathbf{v}_{en}^{n}$), which is in units of m/s, to the change in the

position state ($\dot{\mathbf{p}}$), which is in units of radians of latitude, longitude and meters. The inverse transform is also used, and is defined as follows:

$$\mathbf{T}_{en} = \left(\mathbf{T}_{ne}\right)^{-1} \tag{2.63}$$

2.5.5.3 Velocity Mechanization. In [23], Titterton and Weston develop the navigation equation to solve for the velocity as a function of the accelerometer triad measurements ($\mathbf{f}_{ib}^{b}$), the body-to-nav DCM ($\mathbf{C}_{b}^{n}$), the sidereal angular rate ($\boldsymbol{\omega}_{ie}^{n}$), the transport rate ($\boldsymbol{\omega}_{en}^{n}$), and local gravity ($\mathbf{g}_{l}^{n}$), all described in previous sections:

$$\dot{\mathbf{v}}_{en}^{n} = \mathbf{C}_{b}^{n}\mathbf{f}_{ib}^{b} - (2\boldsymbol{\omega}_{ie}^{n} + \boldsymbol{\omega}_{en}^{n}) \times \mathbf{v}_{en}^{n} + \mathbf{g}_{l}^{n} \tag{2.64}$$

To mechanize this relationship with measurements and calculated values, Equation (2.64) becomes:

$$\tilde{\dot{\mathbf{v}}}_{en}^{n} = \tilde{\mathbf{C}}_{b}^{n}\overline{\mathbf{f}}_{ib}^{b} - (2\tilde{\boldsymbol{\omega}}_{ie}^{n} + \tilde{\boldsymbol{\omega}}_{en}^{n}) \times \tilde{\mathbf{v}}_{en}^{n} + \tilde{\mathbf{g}}_{l}^{n} \tag{2.65}$$

For small time steps, Euler integration may be used to propagate both the position and velocity states as follows:

$$\mathbf{y}(t+\delta t) = \mathbf{y}(t) + \dot{\mathbf{y}}(t)\delta t \tag{2.66}$$

This approximation cannot be used for the attitude state however. Attitude propagation requires Equation (2.51).

2.5.5.4 *Navigation State.* The Navigation State Vector ($\mathbf{x}$) is comprised of the 9 modeled states of interest, that have been described up to this point: position, velocity and attitude [23].

$$\mathbf{x} = \begin{bmatrix} \mathbf{p} \\ \mathbf{v}_{eb}^{n} \\ \Psi_{nb}^{n} \end{bmatrix} = \begin{bmatrix} L & l & h & \vdots & v_N & v_E & v_D & \vdots & \phi & \theta & \psi \end{bmatrix}^{\mathsf{T}} \tag{2.67}$$

The Euler attitude angles (Ψ_{nb}^{n}) are extracted from $\mathbf{C}_{b}^{n}$ in the following manner [26]:

$$\Psi_{nb}^{n} = \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix} = \begin{bmatrix} \arctan\left(\dfrac{C_{b\,23}^{n}}{C_{b\,33}^{n}}\right) \\ \arcsin\left(-C_{b\,13}^{n}\right) \\ \arctan\left(\dfrac{C_{b\,12}^{n}}{C_{b\,11}^{n}}\right) \end{bmatrix} \tag{2.68}$$

where $C_{b\,xy}^{n}$ is the y^{th} element in the x^{th} row of $\mathbf{C}_{b}^{n}$.

2.5.6 Error States. Typical INS applications use the previously described mechanization to generate estimates for the navigation state ($\hat{\mathbf{x}}(t)$). Due to errors in sensor measurements and algorithmic imperfections, this estimate diverges from the true

state. Generally, a state estimate can be modeled as the sum of the true state ($\mathbf{x}(t)$) and the error state ($\delta\mathbf{x}(t)$) [23].

$$\hat{\mathbf{x}}(t) = \mathbf{x}(t) + \delta\mathbf{x}(t) \tag{2.69}$$

In the case of inertial navigation, the relationship between error states and whole states is more complicated. To begin with, a 15-state error vector ($\delta\mathbf{x}(t)$) is defined by the following relationships:

$$\delta\mathbf{x} = \begin{bmatrix} \delta\mathbf{p} \\ \delta\mathbf{v}^n_{eb} \\ \mathbf{e}^n_{nb} \\ \mathbf{a}^b \\ \mathbf{b}^b \end{bmatrix} \tag{2.70}$$

where $\delta\mathbf{p}$ and $\delta\mathbf{v}^n_{eb}$ are the errors in position and velocity, $\mathbf{a}^b$ and $\mathbf{b}^b$ are the accelerometer and gyro biases respectively, and $\mathbf{e}^n_{nb}$ is a vector of angular errors about each of the three *n-frame* axes.

$$\mathbf{e}^n_{nb} = \begin{bmatrix} \varepsilon_x & \varepsilon_y & \varepsilon_z \end{bmatrix}^\mathsf{T} \tag{2.71}$$

The error state dynamics can be expressed as a linear, stochastic state-space model driven by white Gaussian noise [14].

$$\delta\dot{\mathbf{x}}(t) = \mathbf{F}(t)\delta\mathbf{x}(t) + \mathbf{G}(t)\mathbf{w}(t) \tag{2.72}$$

where $\mathbf{F}(t)$ and $\mathbf{G}(t)$ are time varying functions of the whole navigation state, and $\mathbf{w}(t)$ is the dynamics driving noise associated with the inertial sensors and their biases.

The component dynamics have already been developed:

From Equation (2.60):

$$\dot{\mathbf{p}} = \mathbf{T}_{ne}\mathbf{v}^{n}_{en} \tag{2.73}$$

From Equation (2.64):

$$\dot{\mathbf{v}}^{n}_{en} = \mathbf{C}^{n}_{b}\mathbf{f}^{b}_{ib} - (2\boldsymbol{\omega}^{n}_{ie} + \boldsymbol{\omega}^{n}_{en}) \times \mathbf{v}^{n}_{en} + \mathbf{g}^{n}_{l} \tag{2.74}$$

From Equation (2.56):

$$\dot{\mathbf{C}}^{n}_{b} = \mathbf{C}^{n}_{b}\Omega^{b}_{ib} - \Omega^{n}_{in}\mathbf{C}^{n}_{b} \tag{2.75}$$

From Equation (2.39):

$$\dot{\mathbf{a}}^{b} = -\frac{1}{\tau_{a}}\mathbf{a}^{b} + \mathbf{w}^{b}_{a_{bias}} \tag{2.76}$$

From Equation (2.42):

$$\dot{\mathbf{b}}^{b} = -\frac{1}{\tau_{b}}\mathbf{b}^{b} + \mathbf{w}^{b}_{b_{bias}} \tag{2.77}$$

The errors are assumed to be small. Therefore, small perturbations about a nominal may be used to analyze the error dynamics [17].

$$\delta\mathbf{p} \approx \hat{\mathbf{p}} - \mathbf{p} \tag{2.78}$$

$$\delta\mathbf{v} \approx \hat{\mathbf{v}} - \mathbf{v} \tag{2.79}$$

$$\hat{\mathbf{C}}^{n}_{b} \approx [\mathrm{I} - (\varepsilon\times)]\mathbf{C}^{n}_{b} \tag{2.80}$$

Substituting these state error definitions into Equations (2.73)-(2.77), produces the desired error dynamics equations.

$$\delta\dot{\mathbf{p}} = \mathbf{T}_{ne}\delta\mathbf{v}^n_{en} + \delta\mathbf{T}_{ne}\mathbf{v}^n_{en} \tag{2.81}$$

Through some manipulation:

$$\delta\mathbf{T}_{ne}\mathbf{v}^n_{en} = \begin{bmatrix} 0 & 0 & -\dfrac{v_N}{R_0^{\,2}} \\ \dfrac{v_E \tan L}{R_0 \cos L} & 0 & -\dfrac{v_e}{R_0^{\,2}\cos L} \\ 0 & 0 & 0 \end{bmatrix} \delta\mathbf{p} \tag{2.82}$$

Define:

$$\mathbf{T}_{\dot{p}p} = \begin{bmatrix} 0 & 0 & -\dfrac{v_N}{R_0^{\,2}} \\ \dfrac{v_E \tan L}{R_0 \cos L} & 0 & -\dfrac{v_e}{R_0^{\,2}\cos L} \\ 0 & 0 & 0 \end{bmatrix} \tag{2.83}$$

Yielding:

$$\delta\dot{\mathbf{p}} = \mathbf{T}_{ne}\delta\mathbf{v}^n_{en} + \mathbf{T}_{\dot{p}p}\delta\mathbf{p} \tag{2.84}$$

Titterton and Weston [25] show that:

$$\delta\dot{\mathbf{v}}^n_{en} = [\mathbf{C}^n_b\mathbf{f}^b\times]\mathbf{e}^n_{nb} + \mathbf{C}^n_b\delta\mathbf{f}^b - (2\boldsymbol{\omega}^n_{ie} + \boldsymbol{\omega}^n_{en})\times\delta\mathbf{v}^n_{en} - (2\delta\boldsymbol{\omega}^n_{ie} + \delta\boldsymbol{\omega}^n_{en})\times\mathbf{v}^n_{en} - \delta\mathbf{g}^n_l \tag{2.85}$$

Through some manipulation, they also show:

$$\delta\dot{\mathbf{v}}^n_{en} = \mathbf{G}_1\delta\mathbf{p} + [\mathbf{G}_2 - 2\Omega^n_{ie} - \Omega^n_{en}]\delta\mathbf{v}^n_{en} + [\mathbf{C}^n_b\mathbf{f}^b\times]\mathbf{e}^n_{nb} + \mathbf{C}^n_b\mathbf{a}^b + \mathbf{C}^n_b\mathbf{w}^b_a \tag{2.86}$$

where

$$\mathbf{G}_1 = \begin{bmatrix} -v_E\left(2\Omega_{ie}\cos L + \dfrac{v_E}{R_0\cos^2 L}\right) & 0 & \dfrac{{v_E}^2\tan L - v_N v_D}{{R_0}^2} \\ 2\Omega_{ie}\left(v_N\cos L - v_D\sin L\right) + \dfrac{v_N v_E}{R_0\cos^2 L} & 0 & -\dfrac{v_E\left(v_N\tan L + v_D\right)}{{R_0}^2} \\ 2\Omega_{ie}v_E\sin L & 0 & \dfrac{{v_N}^2 + {v_E}^2}{{R_0}^2} \end{bmatrix} \tag{2.87}$$

$$\mathbf{G}_2 = \begin{bmatrix} \dfrac{v_D}{R_0 + h} & -\dfrac{v_E\tan L}{R_N + h} & 0 \\ 0 & -\dfrac{v_N\tan L + v_D}{R_0 + h} & 0 \\ -\dfrac{v_N}{R_N + h} & -\dfrac{v_E}{R_E + h} & 0 \end{bmatrix} \tag{2.88}$$

The matrices $\mathbf{G}_1$ and $\mathbf{G}_2$ account for the errors in the in local gravity ($\delta\mathbf{g}^n_l$), due to the errors in position and velocity.

Titterton and Weston [25] develop the final dynamics equation needed to express the error in attitude, resolved in the *n-frame*:

$$\dot{\varepsilon} = -\Omega^n_{in}\varepsilon - \mathbf{C}^n_b\mathbf{b}^b - \mathbf{C}^n_b\mathbf{w}^b_{b_{bias}} + \delta\boldsymbol{\omega}^n_{in} \tag{2.89}$$

where

$$\delta\boldsymbol{\omega}_{in}^{n} = \frac{\delta\boldsymbol{\omega}_{in}^{n}}{\delta\mathbf{P}}\delta\mathbf{P} + \frac{\delta\boldsymbol{\omega}_{in}^{n}}{\delta\mathbf{v}}\delta\mathbf{v} + \cancel{\frac{\delta\boldsymbol{\omega}_{in}^{n}}{\delta\varepsilon}\delta\varepsilon}, \qquad \frac{\delta\boldsymbol{\omega}_{in}^{n}}{\delta\varepsilon}\delta\varepsilon \approx 0$$

$$\delta\boldsymbol{\omega}_{in}^{n} = \mathbf{E}_{p}\delta\mathbf{P} + \mathbf{E}_{v}\delta\mathbf{v} \tag{2.90}$$

$\boldsymbol{\omega}_{in}^{n}$ is not a function of ε, allowing the term $\frac{\delta\boldsymbol{\omega}_{in}^{n}}{\delta\varepsilon}\delta\varepsilon$ to be dropped. Taking ω_{in}^{n} from Equation (2.54):

$$\mathbf{E}_{p} = \begin{bmatrix} -\Omega_{ie}\sin L & 0 & -\dfrac{{v_E}^2}{\left(R_E + h\right)^2} \\ 0 & 0 & \dfrac{v_N}{\left(R_N + h\right)^2} \\ -\Omega_{ie}\cos L - \dfrac{v_E}{\left(R_N + h\right)\cos^2 L} & 0 & \dfrac{v_E \tan L}{\left(R_N + h\right)^2} \end{bmatrix} \tag{2.91}$$

$$\mathbf{E}_{v} = \begin{bmatrix} 0 & \dfrac{1}{\left(R_E + h\right)} & 0 \\ -\dfrac{1}{\left(R_N + h\right)} & 0 & 0 \\ 0 & -\dfrac{\tan L}{\left(R_N + h\right)} & 0 \end{bmatrix} \tag{2.92}$$

To satisfy the dynamics model from Equation (2.72), the driving noise vector ($\mathbf{w}(t)$) must include the sensor measurement and bias driving noises. It is defined as:

$$\mathbf{w}(t) = \begin{bmatrix} \mathbf{w}_{a}^{b} \\ \mathbf{w}_{b}^{b} \\ \mathbf{w}_{a_{bias}}^{b} \\ \mathbf{w}_{b_{bias}}^{b} \end{bmatrix} \tag{2.93}$$

Combining Equations (2.84), (2.86), (2.91), (2.76) and (2.77), the overall error dynamics equation is obtained for the *n-frame*.

$$\delta\dot{\mathbf{x}}(t) = \mathbf{F}[t;\hat{\mathbf{x}}(t)]\cdot\delta\mathbf{x}(t) + \mathbf{G}(t)\cdot\mathbf{w}(t)$$

$$\delta\dot{\mathbf{x}}(t) = \begin{bmatrix} \mathbf{T}_{\dot{p}p} & \mathbf{T}_{ne} & 0_3 & 0_3 & 0_3 \\ \mathbf{G}_1 & [\mathbf{G}_2 - 2\Omega_{ie}^n - \Omega_{en}^n] & \mathbf{C}_b^n\mathbf{f}^b\times & \mathbf{C}_b^n & 0_3 \\ \mathbf{E}_p & \mathbf{E}_v & -\Omega_{in}^n & 0_3 & -\mathbf{C}_b^n \\ 0_3 & 0_3 & 0_3 & -\frac{1}{\tau_a}\mathrm{I}_3 & 0_3 \\ 0_3 & 0_3 & 0_3 & 0_3 & -\frac{1}{\tau_b}\mathrm{I}_3 \end{bmatrix}_{15x15} \delta\mathbf{x}(t)$$

$$+ \begin{bmatrix} 0_3 & 0_3 & 0_3 & 0_3 \\ \mathbf{C}_b^n & 0_3 & 0_3 & 0_3 \\ 0_3 & -\mathbf{C}_b^n & 0_3 & 0_3 \\ 0_3 & 0_3 & \mathrm{I}_3 & 0_3 \\ 0_3 & 0_3 & 0_3 & \mathrm{I}_3 \end{bmatrix}_{15x12} \mathbf{w}(t) \qquad (2.94)$$

In many cases, the Earth can be modeled as flat and non-rotating. This approximation is acceptable if the rate gyros cannot pick up the Earth's rotation and if the navigation is constrained to a relatively small area. This approximation allows many terms to be dropped, simplifying Equation (2.94) and reducing the computational load.

To model the Earth as flat and non-rotating, $\Omega_{ie} \to 0$ and $R_0 = R_N = R_E \to \infty$. This reduces Equation (2.94) to:

$$\delta\dot{\mathbf{x}}(t) = \begin{bmatrix} 0_3 & \mathbf{T}_{ne} & 0_3 & 0_3 & 0_3 \\ 0_3 & 0_3 & \mathbf{C}_b^n \mathbf{f}^b \times & \mathbf{C}_b^n & 0_3 \\ 0_3 & 0_3 & 0_3 & 0_3 & -\mathbf{C}_b^n \\ 0_3 & 0_3 & 0_3 & -\frac{1}{\tau_a} \mathrm{I}_3 & 0_3 \\ 0_3 & 0_3 & 0_3 & 0_3 & -\frac{1}{\tau_b} \mathrm{I}_3 \end{bmatrix}_{15x15} \delta\mathbf{x}(t) + \begin{bmatrix} 0_3 & 0_3 & 0_3 & 0_3 \\ \mathbf{C}_b^n & 0_3 & 0_3 & 0_3 \\ 0_3 & -\mathbf{C}_b^n & 0_3 & 0_3 \\ 0_3 & 0_3 & \mathrm{I}_3 & 0_3 \\ 0_3 & 0_3 & 0_3 & \mathrm{I}_3 \end{bmatrix}_{15x12} \mathbf{w}(t) \tag{2.95}$$

Note: $\mathbf{T}_{ne}$ cannot be modeled to have $R_0 = R_N = R_E \to \infty$ because it is needed to convert velocity units into degrees of latitude and longitude. This transform remains in the simplified error equation.

The basis for INS mechanization and the associated error model have been laid. The first nine error states are the well understood Pinson error states [27], allowing a feedback extended Kalman filter to be used for aiding. The Kalman filter used to aid the INS, can now be discussed.

2.6 Kalman Filtering

Kalman filtering is the backbone of INS aiding. It provides an optimal estimate of the desired state vector, based on imperfect measurements that indirectly reflect the true state values. An extended Kalman filter is used herein to aid the INS through bearings-only measurements. A development of the EKF follows to aid understanding of the eventual algorithm. This section presents the stochastic modeling and Kalman filtering concepts outlined by Maybeck [14][15].

2.6.1 Linear Stochastic System Model. The linear stochastic system model is essentially the linear system model (Equation (2.5)) with an additional stochastic term to account for the uncertainty of the model dynamics. The general form of the linear stochastic system model can be written in the form of a linear stochastic differential equation [14].

$$\dot{\mathbf{x}} = \mathbf{F}\mathbf{x} + \mathbf{B}\mathbf{u} + \mathbf{G}\mathbf{w} \tag{2.96}$$

where $\mathbf{x}$ is again the state vector, but it is now a vector of random variables. $\mathbf{u}$ is the deterministic input, $\mathbf{F}$ and $\mathbf{B}$ are the system dynamics and input matrices respectively. $\mathbf{G}$ is the noise transformation matrix, and $\mathbf{w}$ is the vector of white, Gaussian, dynamics driving noises. $\mathbf{w}$ is a zero-mean Gaussian process with a covariance kernal, defined by:

$$\mathbf{E}\left\{\mathbf{w}(t)\mathbf{w}^{\mathsf{T}}(t+\tau)\right\} = \mathbf{Q}(t)\delta(\tau) \tag{2.97}$$

where $\mathbf{Q}(t)$ is the noise strength matrix, and $\delta(\tau)$ is the Dirac delta function.

The solution to stochastic differential Equation (2.96) is a Gaussian random vector itself. The dynamics driving noise creates uncertainty in the solution to an otherwise trivial problem. The solution is defined by its mean and covariance [6].

The mean of the random vector $\mathbf{x}(t)$, for $t \in [t_0, \infty)$ is defined as follows:

$$\mathbf{m}_{\mathbf{x}}(t) = \mathbf{E}\left\{\mathbf{x}(t)\right\} = \Phi(t,t_0)\mathbf{E}\left\{\mathbf{x}(t_0)\right\} + \int_{t_0}^{t} \Phi(t,\tau)\mathbf{B}(\tau)\mathbf{u}(\tau)\partial\tau \tag{2.98}$$

with the initial condition

$$\mathbf{E}\left\{\mathbf{x}(t_0)\right\} = \mathbf{x}_0 \tag{2.99}$$

Note that $\Phi(t,t_0)$ is derived in the same manner as in Section 2.4.1 and Equation (2.7). The addition of the driving noise has no effect on $\Phi(t,t_0)$ nor the mean, which is equivalent to the deterministic solution from Equation (2.6) [14].

The covariance of the solution is defined to be

$$\mathbf{P}_{\mathbf{xx}}(t) = \mathbf{E}\left\{\mathbf{x}(t)\mathbf{x}^{\mathsf{T}}(t)\right\} - \mathbf{m}_{\mathbf{x}}(t)\mathbf{m}_{\mathbf{x}}(t)^{\mathsf{T}} \tag{2.100}$$

and is found by solving

$$\mathbf{P}_{\mathbf{xx}}(t) = \Phi(t,t_0)\mathbf{P}_{\mathbf{xx}}(t_0)\Phi^{\mathsf{T}}(t,t_0) + \int_{t_0}^{t} \Phi(t,\tau)\mathbf{G}(\tau)\mathbf{Q}(\tau)\mathbf{G}^{\mathsf{T}}(\tau)\Phi^{\mathsf{T}}(t,\tau)\partial\tau \tag{2.101}$$

with the initial condition

$$\mathbf{P}_{\mathbf{xx}}(t_0) = \mathbf{P}_0 \tag{2.102}$$

Due to the Gaussian nature of the solution, the computed mean provides the best estimate for the true state:

$$\hat{\mathbf{x}}(t) = \mathbf{m}_{\mathbf{x}}(t) \tag{2.103}$$

The computed covariance describes the uncertainty of the estimate. The subscript will be dropped henceforth.

$$\mathbf{P}(t) = \mathbf{P}_{\mathbf{xx}}(t) \tag{2.104}$$

2.6.2 Linear Measurement Model. In order for a Kalman filter to be of use, measurements of the states must be taken. Due to their very nature, measurements are often discrete. The continuous measurement case will not be covered. In the linear stochastic case, measurements of the states are corrupted by additive, white Gaussian noise [14].

$$\mathbf{z}(t_i) = \mathbf{H}(t_i)\mathbf{x}(t_i) + \mathbf{v}(t_i) \tag{2.105}$$

where $\mathbf{z}(t_i)$ is a vector of measurements taken at a specific time instance $(t = t_i)$. $\mathbf{H}(t_i)$ is the observation matrix, relating the states to the measurements produced by the sensor, and $\mathbf{v}(t_i)$ is the sensor corruption noise vector, with covariance kernel defined as:

$$\mathbf{E}\left\{\mathbf{v}(t_i)\mathbf{v}^{\mathsf{T}}(t_j)\right\} = \mathbf{R}(t_i)\delta_{ij} \tag{2.106}$$

where $\mathbf{R}(t_i)$ is the covariance matrix for the corruption noise, and δ_{ij} is the Kroeneker delta function.

2.6.3 Linear Kalman Filter. The Kalman filter allows the use of imperfect measurements, to estimate the solution to the stochastic differential Equation (2.96). Without measurements, the covariance of the solution would grow unbounded, rendering the state estimate meaningless to a user. Kalman filtering bounds the covariance, or uncertainty in the estimate, but does not provide perfect knowledge. In the linear case, the Gauss-Markov nature of the state model allows the use a simple recursive cycle, repeating for every measurement.

2.6.3.1 Kalman Filter Cycle. The Kalman filter cycle has two parts: the propagation phase, and the update step. In the propagation phase, the Kalman filter estimates the new mean and covariance of the state estimate for some time passage. During the update step, the Kalman filter adjusts the propagated mean and covariance based on measurements taken in. This cycle is repeated for every measurement. Figure 2.6 illustrates this cycle [6][14].

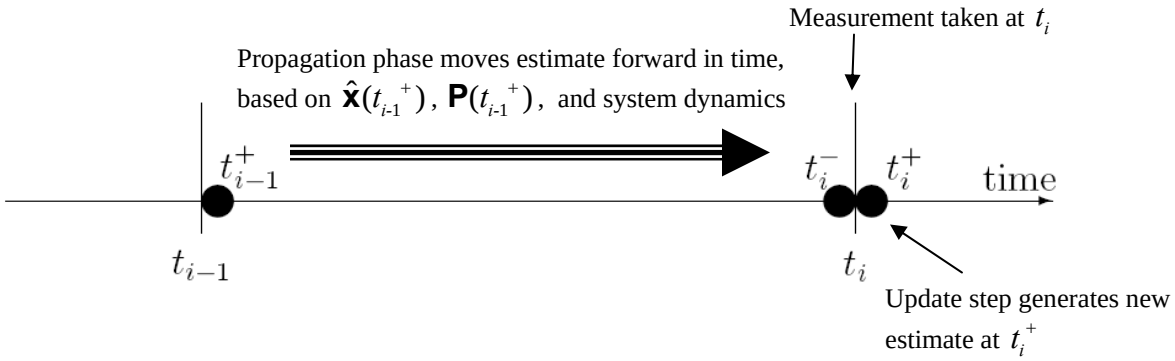

Figure 2.6: Kalman filter cycle.

Note the convention used to indicate time immediately before and after a measurement (illustrated in Figure 2.6, and explained below):

- t_i refers to an instance in time; typically when a measurement is taken.
- t_i^- refers to the very instant before t_i. $\hat{\mathbf{x}}(t_{i-1}^-)$ is the Kalman filter estimate the instant the measurement is taken, but just before it is applied. $\hat{\mathbf{x}}(t_{i-1}^-)$ and $\mathbf{P}(t_{i-1}^-)$ are the output of the propagation phase.
- t_i^+ refers to the very instant after a measurement is taken, and the update step is applied. $\hat{\mathbf{x}}(t_{i-1}^+)$ and $\mathbf{P}(t_{i-1}^+)$ are the output of the update step.

2.6.3.2 Propagation Equations. The state estimate and covariance are propagated through time using the solution to the stochastic differential Equation (2.96) shown in Equations (2.98) and (2.101) [14]. Applying the Kalman filter cycle convention, the propagation phase becomes:

$$\hat{\mathbf{x}}(t_i^-) = \Phi(t_i, t_{i-1})\hat{\mathbf{x}}(t_{i-1}^+) + \int_{t_{i-1}}^{t_i} \Phi(t_i, \tau)\mathbf{B}(\tau)\mathbf{u}(\tau)\partial\tau \tag{2.107}$$

$$\mathbf{P}(t_i^-) = \Phi(t_i, t_{i-1})\mathbf{P}(t_{i-1}^+)\Phi^{\mathsf{T}}(t_i, t_{i-1}) + \int_{t_{i-1}}^{t_i} \Phi(t_i, \tau)\mathbf{G}(\tau)\mathbf{Q}(\tau)\mathbf{G}^{\mathsf{T}}(\tau)\Phi^{\mathsf{T}}(t_i, \tau)\partial\tau \tag{2.108}$$

In the discrete time case:

$$\hat{\mathbf{x}}(t_i^-) = \Phi(t_i, t_{i-1})\hat{\mathbf{x}}(t_{i-1}^+) + \mathbf{B}_{\mathbf{d}}(t_{i-1})\mathbf{u}(t_{i-1}) \tag{2.109}$$

$$\mathbf{P}(t_i^-) = \Phi(t_i, t_{i-1})\mathbf{P}(t_{i-1}^+)\Phi^{\mathsf{T}}(t_i, t_{i-1}) + \mathbf{G}_{\mathbf{d}}(t_{i-1})\mathbf{Q}_{\mathbf{d}}(t_{i-1})\mathbf{G}_{\mathbf{d}}{}^{\mathsf{T}}(t_{i-1}) \tag{2.110}$$

where $\mathbf{u}(t)$ is assumed constant over the interval $t \in [t_{i-1}, t_i)$, and :

$$\mathbf{B_d}(t_{i-1}) = \int_{t_{i-1}}^{t_i} \Phi(t_i, \tau)\mathbf{B}(\tau)\partial\tau \tag{2.111}$$

$$\mathbf{G_d}(t_{i-1})\mathbf{Q_d}(t_{i-1})\mathbf{G_d}^{\mathsf{T}}(t_{i-1}) = \int_{t_{i-1}}^{t_i} \Phi(t_i, \tau)\mathbf{G}(\tau)\mathbf{Q}(\tau)\mathbf{G}^{\mathsf{T}}(\tau)\Phi^{\mathsf{T}}(t_i, \tau)\partial\tau \tag{2.112}$$

2.6.3.3 Update Equations. During the update step, the state estimate and covariance are adjusted to account for the measurement coming in. The Kalman filter gain appropriately weighs the measurement against the estimate, as a function of the system model [14].

Kalman filter gain:

$$\mathbf{K}(t_i) = \mathbf{P}(t_i^-)\mathbf{H}^{\mathsf{T}}(t_i)\left[\mathbf{H}(t_i)\mathbf{P}(t_i^-)\mathbf{H}^{\mathsf{T}}(t_i) + \mathbf{R}(t_i)\right]^{-1} \tag{2.113}$$

Update Equations:

$$\hat{\mathbf{x}}(t_i^+) = \hat{\mathbf{x}}(t_i^-) + \mathbf{K}(t_i)\left[\mathbf{z}(t_i) - \mathbf{H}(t_i)\hat{\mathbf{x}}(t_i^-)\right] \tag{2.114}$$

$$\mathbf{P}(t_i^+) = \mathbf{P}(t_i^-) - \mathbf{K}(t_i)\mathbf{H}(t_i)\mathbf{P}(t_i^-) \tag{2.115}$$

2.6.4 Non-Linear Stochastic System Model. In many cases, a linear system model does not suffice. The non-linear stochastic system model, used in this research, is a variation on the deterministic non-linear system from Equation (2.13). Like the linear model, it has an additional, dynamics driving, white Gaussian noise term [15].

$$\dot{\mathbf{x}}(t) = \mathbf{f}[\mathbf{x}(t), \mathbf{u}(t), t] + \mathbf{G}(t)\mathbf{w}(t) \tag{2.116}$$

where $\mathbf{G}(t)\mathbf{w}(t)$ is defined as in the linear case; Equations (2.96), (2.97).

2.6.5 Non-Linear Measurement Model. The non-linear measurement model takes on a similar form [15].

$$\mathbf{z}(t_i) = \mathbf{h}[\mathbf{x}(t_i), t_i] + \mathbf{v}(t_i) \tag{2.117}$$

where $\mathbf{h}[\mathbf{x}(t_i), t_i]$ is a non-linear function of $\mathbf{x}(t_i)$ and time, and $\mathbf{v}(t_i)$ is defined as in the linear case; Equation (2.105).

2.6.6 Linearized Model. The non-linear stochastic system model massively complicates the estimation process. The solution to the stochastic differential is no longer trivial, and in some cases, unsolvable. Fortunately, many non-linear systems can be modeled as linear for small perturbations about a nominal trajectory. The extended Kalman filter takes advantage of this property.

Because the dynamics driving noise is additive, it does not complicate the linearization process. The non-linear stochastic system may be approximated to be linear about a nominal trajectory as described in Section 2.4.2 [15].

$$\dot{\hat{\mathbf{x}}}(t) + \delta\dot{\mathbf{x}}(t) = \mathbf{f}[\hat{\mathbf{x}}(t), \mathbf{u}(t), t] + \mathbf{F}[t; \hat{\mathbf{x}}(t), \mathbf{u}(t)]\delta\mathbf{x}(t) + \mathbf{G}(t)\mathbf{w}(t) \tag{2.118}$$

$$\delta\dot{\mathbf{x}}(t) = \mathbf{F}[t; \hat{\mathbf{x}}(t), \mathbf{u}(t)]\delta\mathbf{x}(t) + \mathbf{G}(t)\mathbf{w}(t) \tag{2.119}$$

$$\mathbf{F}[\mathbf{t};\hat{\mathbf{x}}(t)] = \left.\frac{\partial \mathbf{f}[\mathbf{x},\mathbf{u}(t),t]}{\partial \mathbf{x}}\right|_{\mathbf{x}=\hat{\mathbf{x}}(t),\mathbf{u}(t),t} = \left.\begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_1} & \cdots & \frac{\partial f_n}{\partial x_n} \end{bmatrix}\right|_{\mathbf{x}=\hat{\mathbf{x}}(t),\mathbf{u}(t),t} \tag{2.120}$$

The resultant system models the small perturbations, or errors about the nominal. This fits well with the error states chosen for the INS. Note that the nominal chosen is the state estimate ($\hat{\mathbf{x}}(t)$). The optimal nominal choice would be the true state, but since the true state is unobtainable, $\hat{\mathbf{x}}(t)$ provides a reasonable approximation [15]. The result is a poorer performing Kalman filter. Additionally, the input vector is assumed to be known ($\delta\mathbf{u}(t) = \mathbf{0}$).

The same process can be applied to the measurement equation [26][15].

$$\mathbf{z}(t_{i+1}) = \mathbf{h}[\hat{\mathbf{x}}(t_{i+1}^{-}), t_{i+1}] + \mathbf{H}[t_{i+1};\hat{\mathbf{x}}(t_{i+1}^{-})]\delta\mathbf{x}(t_{i+1}) + \mathbf{v}(t_{i+1}) \tag{2.121}$$

$$\delta\mathbf{z}(t_{i+1}) = \mathbf{H}[t_{i+1};\hat{\mathbf{x}}(t_{i+1})]\delta\mathbf{x}(t_{i+1}) + \mathbf{v}(t_{i+1}) \tag{2.122}$$

$$\mathbf{H}[t_i;\hat{\mathbf{x}}(t_i^{-})] = \left.\frac{\partial \mathbf{h}[\mathbf{x},t_i]}{\partial \mathbf{x}}\right|_{\mathbf{x}=\hat{\mathbf{x}}(t_i^{-}),t_i} = \left.\begin{bmatrix} \frac{\partial h_1}{\partial x_1} & \cdots & \frac{\partial h_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial h_m}{\partial x_1} & \cdots & \frac{\partial h_m}{\partial x_n} \end{bmatrix}\right|_{\mathbf{x}=\hat{\mathbf{x}}(t_i^{-}),t_i} \tag{2.123}$$

2.6.7 Extended Kalman Filter. After linearization, the system error model fits the linear form needed to create a linear Kalman filter. Maybeck develops the extended Kalman filter (EKF) fully, yielding the propagation and update equations as a function of the linearization gradients ($\mathbf{F}[t;\hat{\mathbf{x}}(t),\mathbf{u}(t)]$ and $\mathbf{H}[t_i;\hat{\mathbf{x}}(t_i^-)]$) [15].

2.6.7.1 EKF Propagation Equations. The EKF estimate is propagated forward to the next time sample t_{i+1} by integrating the following over the interval $t \in [t_i, t_{i+1}]$:

$$\dot{\hat{\mathbf{x}}}(t/t_i) = \mathbf{f}[\hat{\mathbf{x}}(t/t_i), \mathbf{u}(t), t] \tag{2.124}$$

$$\dot{\mathbf{P}}(t/t_i) = \mathbf{F}[t;\hat{\mathbf{x}}(t/t_i),\mathbf{u}(t)]\mathbf{P}(t/t_i) + \mathbf{P}(t/t_i)\mathbf{F}^{\mathbf{T}}[t;\hat{\mathbf{x}}(t/t_i),\mathbf{u}(t)] + \mathbf{G}(t)\mathbf{Q}(t)\mathbf{G}^{\mathbf{T}}(t) \tag{2.125}$$

with the initial conditions:

$$\hat{\mathbf{x}}(t_i/t_i) = \hat{\mathbf{x}}(t_i^+) \tag{2.126}$$

$$\mathbf{P}(t_i/t_i) = \mathbf{P}(t_i^+) \tag{2.127}$$

The propagation produces:

$$\hat{\mathbf{x}}(t_{i+1}^-) = \hat{\mathbf{x}}(t_{i+1}/t_i) \tag{2.128}$$

$$\mathbf{P}(t_{i+1}^-) = \mathbf{P}(t_{i+1}/t_i) \tag{2.129}$$

2.6.7.2 EKF Update Equations. The EKF update is accomplished by the following equations [15]:

$$\mathbf{K}(t_i) = \mathbf{P}(t_i^-)\mathbf{H}^{\mathbf{T}}[t_i;\hat{\mathbf{x}}(t_i^-)]\left\{\mathbf{H}[t_i;\hat{\mathbf{x}}(t_i^-)]\mathbf{P}(t_i^-)\mathbf{H}^{\mathbf{T}}[t_i;\hat{\mathbf{x}}(t_i^-)] + \mathbf{R}(t_i)\right\}^{-1} \tag{2.130}$$

$$\delta\hat{\mathbf{x}}(t_i^+) = \delta\hat{\mathbf{x}}(t_i^-) + \mathbf{K}(t_i)\left\{\mathbf{z}(t_i) - \mathbf{h}[\hat{\mathbf{x}}(t_i^-), t_i] - \mathbf{H}[t_i; \hat{\mathbf{x}}(t_i^-)]\delta\mathbf{x}(t_i)\right\} \tag{2.131}$$

$$\mathbf{P}(t_i^+) = \mathbf{P}(t_i^-) - \mathbf{K}(t_i)\mathbf{H}[t_i; \hat{\mathbf{x}}(t_i^-)]\mathbf{P}(t_i^-) \tag{2.132}$$

where $\delta\hat{\mathbf{x}}(t_i)$ is the error state vector containing the Pinson errors states.

2.6.8 Extended Kalman Filter and INS Integration. Figure 2.7 illustrates how the EKF and INS work together to bound the error growth.

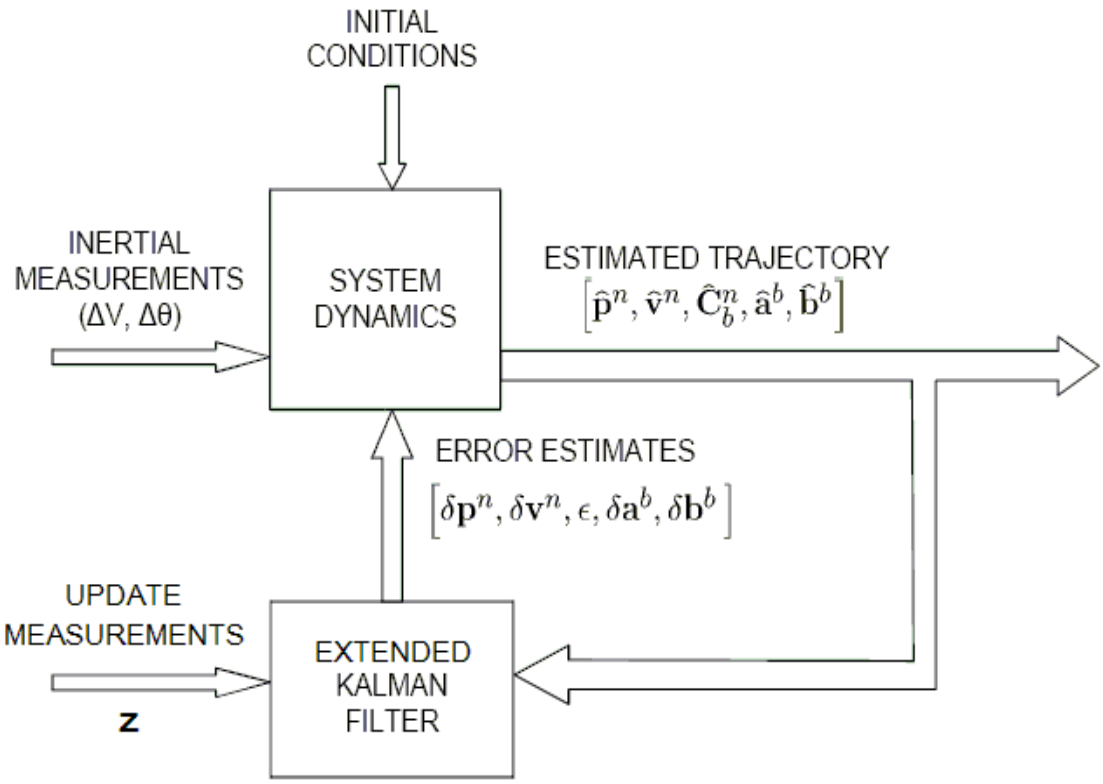

Figure 2.7: Extended Kalman Filter and INS integration. The EKF receives the state estimate from the INS system dynamics, as well as measurements from the update sensors. The EKF produces an error state estimate, and applies it as a correction to the INS [26].

2.6.9 Implicit Measurement Equation. Sometimes, the measurement model used does not fit the convenient form in Equation (2.117). Non-linearities, or singular matrices may not allow one to isolate the measurement vector on the left hand side of the equation. In this case, the implicit measurement model may be used [19].

$$\mathbf{0} = \mathbf{g}(\mathbf{x}, \mathbf{y}) \tag{2.133}$$

$$\mathbf{z} = \mathbf{y} + \mathbf{v}, \quad \mathbf{v} \sim N(\mathbf{0}, \mathbf{R}) \tag{2.134}$$

$$\mathbf{g}(\mathbf{x}, \mathbf{y}) \approx \mathbf{g}(\hat{\mathbf{x}}, \mathbf{z}) + \left.\frac{\partial \mathbf{g}}{\partial \mathbf{x}}\right|_{\hat{\mathbf{x}}} \delta\mathbf{x} + \left.\frac{\partial \mathbf{g}}{\partial \mathbf{y}}\right|_{\mathbf{z}} (\hat{\mathbf{y}} - \mathbf{z}) + \mathbf{H.O.T.} \tag{2.135}$$

where $\mathbf{g}(\mathbf{x}, \mathbf{y})$ is the functional relationship between the true state $\mathbf{x}$ and the perfect measurement $\mathbf{y}$. A new measurement model can be created:

$$\mathbf{g}(\hat{\mathbf{x}}, \mathbf{z}) = \left.\frac{\partial \mathbf{g}}{\partial \mathbf{x}}\right|_{\hat{\mathbf{x}}} \delta\mathbf{x} + \mathbf{v}', \quad \mathbf{v}' \sim N\left(\mathbf{0}, \mathbf{R}'\right) \tag{2.136}$$

$$\mathbf{R}' = \left(\left.\frac{\partial \mathbf{g}}{\partial \mathbf{y}}\right|_{\mathbf{z}}\right) \mathbf{R} \left(\left.\frac{\partial \mathbf{g}}{\partial \mathbf{y}}\right|_{\mathbf{z}}\right)^{\mathsf{T}} \tag{2.137}$$

where $\mathbf{g}(\hat{\mathbf{x}}, \mathbf{z})$ is substituted for $\mathbf{z}$ in the EKF update Equation (2.131), and $\mathbf{R}'$ is substituted for $\mathbf{R}$ in the EKF gain Equation (2.130). Lastly, $\mathbf{H}[t_i; \hat{\mathbf{x}}(t_i^-)]$ is replaced with $\left.\frac{\partial \mathbf{g}}{\partial \mathbf{x}}\right|_{\hat{\mathbf{x}}}$. The rest of the EKF remains unchanged.

The extended Kalman filter will be used to aid the INS using bearings only measurements from imagery. The optics involved with taking images can now be discussed.

2.7 Imaging

In order to aid INS with imagery, the mathematics of optics must be understood. This section presents the aspects of this broad field which will be applied to the research herein, to include optical projection theory, sensor models, measurement equations, and feature tracking.

2.7.1 Optical Projection Theory. Optical projection theory relates the real world geometry to the geometry of an image passed through a lens. The thin lens model relates the image to the true scene according to Figure 2.8, and the following equation [10]:

$$\frac{1}{Z} + \frac{1}{z} = \frac{1}{f} \tag{2.138}$$

where Z and z are the distances of the object and image to the lens respectively, and f is the focal length of the thin lens.

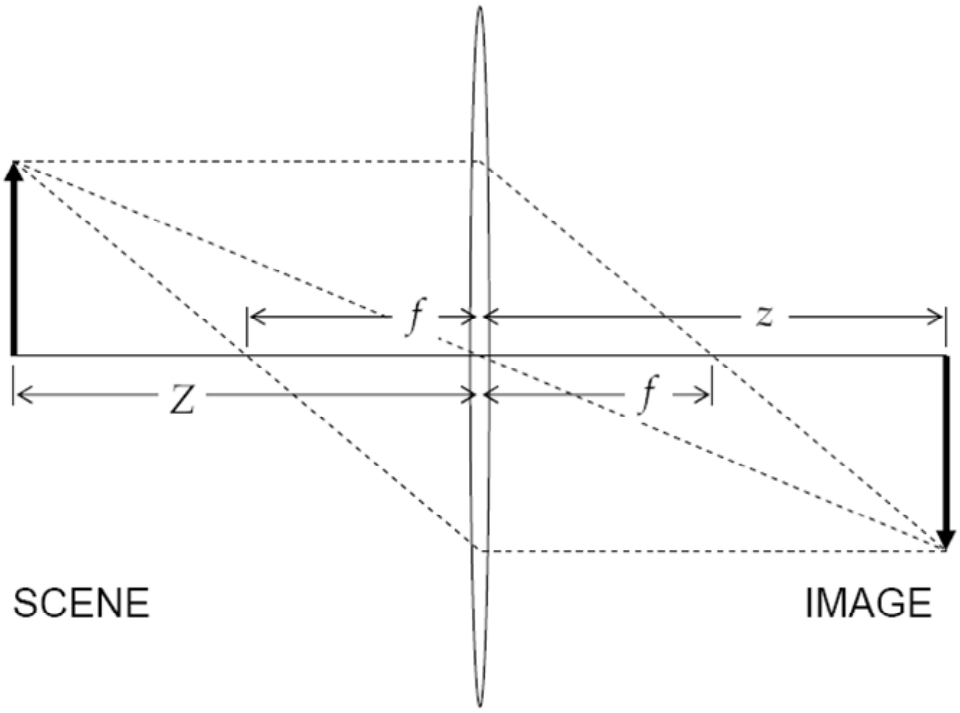

Figure 2.8: Thin lens camera model. The thin lens model directs parallel light rays toward the focus, resulting in an image. Figure is not to scale [26].

Decreasing the aperture of the thin lens to a theoretical zero allows the use of a pinhole camera model [10]. The pinhole camera model relates the size of an image, as it falls onto an image plane, to the size of the true object, as given in Equation (2.138) and illustrated in Figure 2.9. In order for a pinhole model to be applied, the "fish eye" distortion created by a wide angle lens must be removed from the image and associated features. A discussion of how this is accomplished is included in [36]. For the purposes of this discussion, fish eye distortion will be assumed to be removed from all images and the pinhole camera model assumed valid.

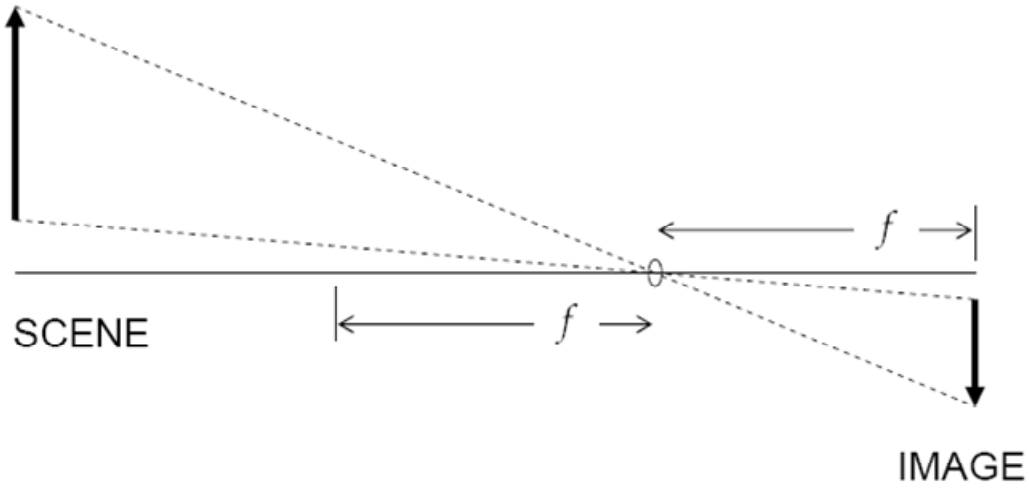

Figure 2.9: Pinhole camera model. The pinhole camera is a theoretical camera model in which a thin lens aperture approaches zero. The projected image is inverted on the image plane [26].

Inverting this image has the effect of righting it, and projecting it one focal length in front of the aperture; see Figure 2.10. Consider a point source object. The projection of the object's image onto the image plane ($\mathbf{s}^{proj}$) is a function of the object's position vector, resolved in the *c-frame* ($\mathbf{s}^{c}$), and the focal length of the camera lens [10].

$$\mathbf{s}^{proj} = \left(\frac{f}{s_z^c} \right) \mathbf{s}^{c} \tag{2.139}$$

where s_z^c is the z-component of $\mathbf{s}^{c}$, or distance of the point source from the aperture, projected onto the *c-frame's* z-axis.

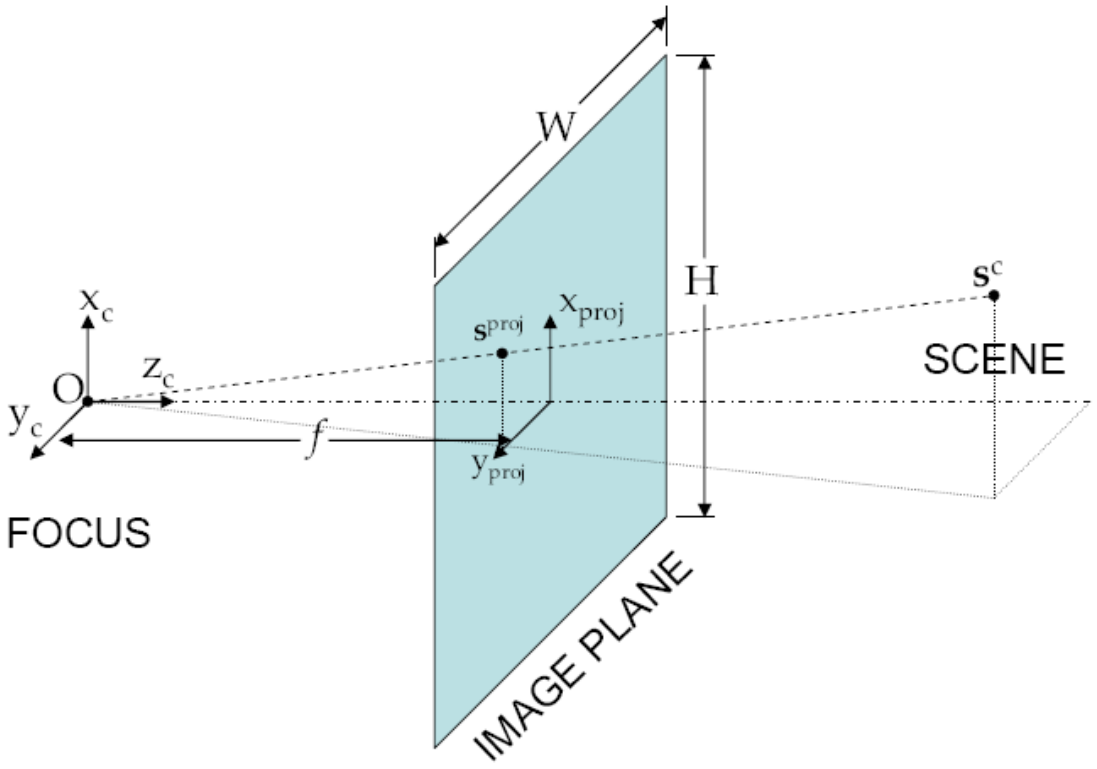

Figure 2.10: Image projection model. A point source object ($\mathbf{s}^c$) is projected onto the image plane of the camera ($\mathbf{s}^{proj}$) [26].

Imaging arrays lie on the image plane in order to capture the focused image. They are made up of a ($M \times N$) array of rectangular pixels. The upper left pixel has the coordinates (1,1). The pixel co-ordinate frame increases down and right [26] (see Figure 2.11).

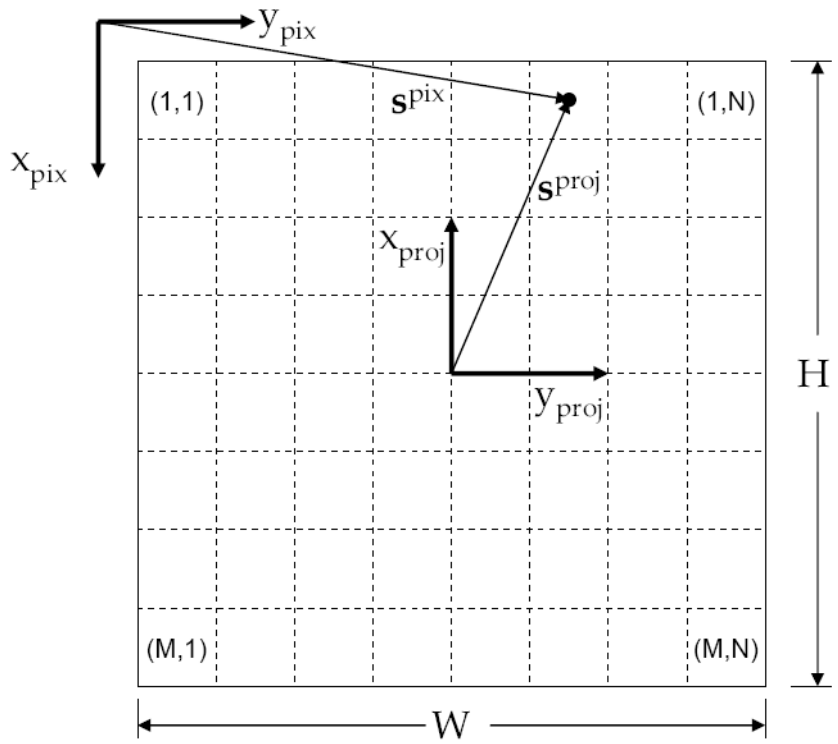

Figure 2.11: Camera image array. The imaging array of the camera consists of an (M x N) array of pixels. The physical height and width of the array are represented by H and W, respectively [26].

For an array with physical dimensions ($H \times W$), the transform to obtain pixel coordinates from the projected image vector is [10]:

$$\mathbf{s}^{pix} = \begin{bmatrix} -\frac{M}{H} & 0 & 0 \\ 0 & \frac{N}{W} & 0 \end{bmatrix} \mathbf{s}^{proj} + \begin{bmatrix} \frac{M+1}{2} \\ \frac{N+1}{2} \end{bmatrix} \tag{2.140}$$

Note that $\mathbf{s}^{pix}$ is a two-dimensional vector of pixel locations. It does not have to take on integer values, however.

Combining Equations (2.139) and (2.140) yields a transform from the camera frame to the pixel plane [26].

$$\underline{\mathbf{z}}_p = \frac{1}{s_z^c}\begin{bmatrix} -f\frac{M}{H} & 0 & \frac{M+1}{2} \\ 0 & f\frac{N}{W} & \frac{N+1}{2} \\ 0 & 0 & 1 \end{bmatrix}\mathbf{s}^c \tag{2.141}$$

$$\underline{\mathbf{z}}_p = \frac{1}{s_z^c}\mathbf{T}_c^{pix}\mathbf{s}^c \tag{2.142}$$

Note that $\underline{\mathbf{z}}_p$ is the homogeneous pixel coordinate vector form of $\mathbf{s}^{pix}$. It is also the measurement produced by the camera.

$\mathbf{T}_c^{pix}$ is the invertible, homogeneous transform matrix between the camera frame and the pixel plane. The inverse is defined:

$$\mathbf{T}_{pix}^c = \left(\mathbf{T}_c^{pix}\right)^{-1} = \begin{bmatrix} -\frac{H}{fM} & 0 & \frac{H(M+1)}{2fM} \\ 0 & \frac{W}{fN} & -\frac{W(N+1)}{2fN} \\ 0 & 0 & 1 \end{bmatrix} \tag{2.143}$$

$$\mathbf{s}^c = s_z^c\mathbf{T}_{pix}^c\underline{\mathbf{z}}_p \tag{2.144}$$

The object's *camera-frame* position vector is thus expressed as the pixel plane measurement, transformed into the *c-frame*, and scaled by the z-component. Given the pixel plane measurement $\underline{\mathbf{z}}_p$, one must estimate the z-component of the object position vector in order to estimate the full object position vector. This can be done many ways, and will be discussed in Chapter 3.

The goal of imaging is to determine the relative position of an object to the camera in the *n-frame* (given the pixel plane vector). Define Δ_{tgt}^{n} to be the relative difference between the imaged/tracked object and the *n-frame* origin (resolved in the *n-frame*). Define $\mathbf{p}_{cam}^{b}$ to be the relative position of the camera origin (resolved in the *b-frame*). Define $\mathbf{C}_{c}^{b}$ to be the DCM corresponding to the camera alignment to the aircraft body. Then Δ_{tgt}^{n} can be related to $\underline{\mathbf{z}}_p$ by the following (see Figure 2.12).

$$\Delta_{tgt}^{n} = \mathbf{C}_{b}^{n}\mathbf{p}_{cam}^{b} + \mathbf{C}_{b}^{n}\mathbf{C}_{c}^{b}\mathbf{s}^{c} \tag{2.145}$$

$$\Delta_{tgt}^{n} = \mathbf{C}_{b}^{n}\mathbf{p}_{cam}^{b} + s_{z}^{c}\mathbf{C}_{c}^{n}\mathbf{T}_{pix}^{c}\underline{\mathbf{z}}_{p} \tag{2.146}$$

Note that $\mathbf{C}_{c}^{b}\mathbf{T}_{pix}^{c}$ is a constant matrix expression for a rigidly mounted camera, and $\mathbf{C}_{c}^{n} = \mathbf{C}_{b}^{n}\mathbf{C}_{c}^{b}$.

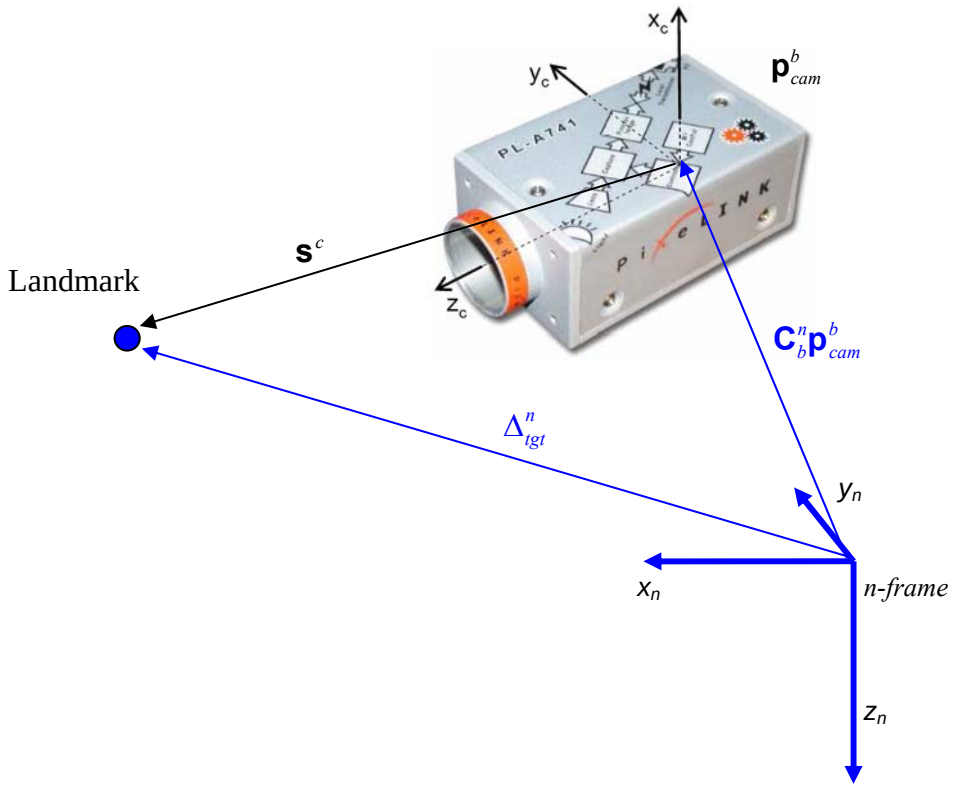

Figure 2.12: Relative landmark geometry. The *n-frame* and *b-frame* share an origin, but are aligned according to the aircraft Euler Angles ($\mathbf{C}^n_b$). The relative position of the tracked object to the *n-frame* (Δ^n_{tgt}) is the sum of the *c-frame* object position vector ($\mathbf{s}^c$) and the camera position inside the aircraft, resolved in the *n-frame* ($\mathbf{C}^n_b\mathbf{p}^b_{cam}$).

Note that Δ^n_{tgt} is a relative vector position in the *n-frame* with units of meters. Δ^n_{tgt} expressed as a function of the aircraft and target geodetic position (latitude, longitude, and height) is:

$$\Delta^n_{tgt} = \mathbf{T}_{en}\left(\mathbf{p}_{tgt} - \mathbf{p}\right) \tag{2.147}$$

where $\mathbf{p}$ and $\mathbf{p}_{tgt}$ are the geodetic positions of the aircraft and target respectively.

Substituting Equation (2.147) into (2.146) yields the measurement equation for this camera model:

$$\underline{\mathbf{z}}_p = \frac{1}{s_z^c}\mathbf{T}_c^{pix}\mathbf{C}_n^c\left(\mathbf{T}_{en}\left(\mathbf{p}_{tgt}-\mathbf{p}\right)-\mathbf{C}_b^n\mathbf{p}_{cam}^b\right) \tag{2.148}$$

2.7.2 Feature Tracking. Aiding INS with imagery can only be accomplished if fixed landmarks on the ground can be tracked from image frame to image frame. Many feature tracking algorithms are described by Veth and others [10][26]. The inner workings of these feature trackers will not be discussed here, merely the characterization of the measurements they provide.

For the purpose of this discussion, landmarks fall into two categories: surveyed landmarks, and landmarks of opportunity (LOO). Surveyed landmarks are those with geodetic positions that are known via some independent source and can be identified in a scene. Landmarks of opportunity have not been surveyed independently and must be located as they are tracked. For the purpose of this discussion, a feature is defined as the projection of a specific landmark onto the image. Feature uniqueness allows successive matches and tracking frame to frame.

Feature trackers aim to track a landmark from frame to frame. These landmarks appear as a collection of pixels that correlate spatially and geometrically between frames. The output of a feature tracker is the homogeneous pixel coordinate vector $\underline{\mathbf{z}}_p$. Ideally, this vector would correspond to the exact same *n-frame* location with every frame;

whether it be the center of mass, or a distinct corner, is moot. Unfortunately, this pixel coordinate vector is plagued by errors.

The first major error is due to the quantization nature of an imaging array [26]. Pixels are discrete, but feature locations aren't necessarily. Some feature generation algorithms claim sub-pixel level precision. This is questionable because such precision is beyond the spatial sampling frequency of the camera. While the feature tracker may interpolate between groups of pixels, the pixels between which they interpolate are discrete. This induces a uniformly distributed spatial error, one pixel wide.

$$\bar{\underline{\mathbf{z}}}_p = \underline{\mathbf{z}}_p + \mathbf{n}_q \tag{2.149}$$

where $\mathbf{n}_q$ is the vector form of the uniform random variable measurement noise, defined as follows. Note that the third element is always zero, as it is a two dimensional error:

$$\mathbf{E}\left\{\mathbf{n}_q\right\} = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^{\mathsf{T}} \tag{2.150}$$

$$\mathbf{E}\left\{\mathbf{n}_q \mathbf{n}_q^{\mathsf{T}}\right\} = \mathbf{R}_q = \begin{bmatrix} \frac{1}{12}\left(\frac{H}{M}\right)^2 & 0 & 0 \\ 0 & \frac{1}{12}\left(\frac{W}{N}\right)^2 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{2.151}$$

Additional noise sources sum together and are modeled as additive, white, and Gaussian in nature.

$$\bar{\underline{\mathbf{z}}}_p = \underline{\mathbf{z}}_p + \mathbf{n}_q + \mathbf{n}_G \tag{2.152}$$

where $\mathbf{n}_G$ is the vector form of the Gaussian random variable measurement noise, defined as:

$$\mathbf{E}\{\mathbf{n}_G\} = [0 \quad 0 \quad 0]^{\mathbf{T}} \tag{2.153}$$

$$\mathbf{E}\left\{\mathbf{n}_G \mathbf{n}_G{}^T\right\} = \mathbf{R}_G = \begin{bmatrix} R_x & R_{xy} & 0 \\ R_{xy} & R_y & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{2.154}$$

where $\mathbf{R}_G$ is the covariance matrix of the white Gaussian noise. If $\mathbf{R}_G$ is much larger than $\mathbf{R}_q$, then they can be modeled as a single, white Gaussian noise, but this is a function of the quality of the tracking algorithm and imaging array used.

2.8 Geometric Dilution of Precision

Non-linear measurements are often functions of geometry. This is especially true in the case of tracking features in successive images. Some geometries are more suited to deliver better performance than others (e.g., as tangents approach 90 degrees, singularities form). A quantifiable measure of quality (as a function of geometry) is called the geometric dilution of precision (GDOP). This scalar value relates the relative

precision or quality of measurement that can be obtained from differing geometric scenarios. The absolute value of GDOP is less insightful than the difference between two GDOP values; a high GDOP means much dilution, whereas a low GDOP means very little dilution [21].

The in the case of the EKF, GDOP is a function of the linearized measurement observation matrix ($\mathbf{H}[t_i;\hat{\mathbf{x}}(t_i^-)]$) [19].

$$\mathbf{GDOP} = \sqrt{\mathbf{tr}\left\{(\mathbf{H}^{\mathbf{T}}[t_i;\hat{\mathbf{x}}(t_i^-)]\mathbf{H}[t_i;\hat{\mathbf{x}}(t_i^-)])^{-1}\right\}} \tag{2.155}$$

High GDOP values indicate that the imprecision of imperfect measurements will be magnified by geometric effects. Fortunately, this knowledge is built into the EKF, and the relative weighting of a measurement experiencing high GDOP is lowered. This effect causes the EKF to behave much like it does with a very uncertain measurement (large $\mathbf{R}(t_i)$) [19]. For this reason, analyzing the GDOP of a system can provide insight into the performance of the associated EKF. High GDOP essentially weakens the aiding strength of a measurement.

2.9 System Observability

A linear system is said to be observable if knowledge of the initial state can be learned after a finite amount of time and a finite number of measurements. This is significant to aiding an INS because learning the state through measurements is the entire purpose. For the time invariant case, the observability matrix is a function of the system dynamics matrix ($\mathbf{F}$) and the measurement matrix ($\mathbf{H}$) [26]; defined:

$$\mathbf{Q} = \begin{bmatrix} \mathbf{H} \\ \mathbf{HF} \\ \vdots \\ \mathbf{HF}^{n-1} \end{bmatrix} \tag{2.156}$$

where n is the dimension of the associated state vector. The $\mathbf{H}\,\mathbf{F}$ pair is said to be fully observable if $\mathbf{Q}$ has rank *n*. This means that the measurement model chosen is sufficient to observe the total state. Because both matrices are functions of the state vector, insight about observability can only be obtained for perturbations about the given state trajectory. In the case of INS, one must examine $\mathbf{Q}$ for all the desired flight profiles and envelopes; no sweeping generalizations may be made.

In the time-varying case, another method of examining the strength of aiding action (provided to the INS by update measurements) must be used. It involves studying the observability grammian. A system is said to be completely observable if the observability grammian ($\mathbf{W}(t)$), for a particular measurement equation, has full rank.

The observability grammian is a function of the system dynamics matrix ($\mathbf{F}$), the observation matrix ($\mathbf{H}$) and the nominal trajectory about with the system is linearized ($\mathbf{x_c}$).

$$\mathbf{W}(t) = \int_0^t e^{\mathbf{F}^T t} \cdot \mathbf{H}^T[\mathbf{x_c}(t)]\mathbf{H}[\mathbf{x_c}(t)]e^{\mathbf{F}t} dt \tag{2.157}$$

The condition number of $\mathbf{W}(t)$ also speaks to the observability of the measurements. A high condition number indicates weak aiding along certain trajectories while strong aiding along others.

Another method of expressing GDOP is with the observability grammian. The GDOP is defined as:

$$\mathbf{GDOP} = \sqrt{\mathbf{trace}\left\{\mathbf{W}(t)^{-1}\right\}} \tag{2.158}$$

2.10 Summary

In summary, this chapter laid out the framework to design an INS and develop an aiding strategy built around the extended Kalman filter. The basic optics and camera reference frame transforms were explored to allow the development of the measurement equation in Chapter 3. Lastly, the tools by which the performance will be predicted and evaluated were presented. Chapter 3 will speak to the specific design used for the flight test.

III. Developing the SLAAMR Algorithm

The concept of simultaneously mapping an environment using imagery and navigating using this map is not new. The Simultaneous Location And Mapping (SLAM) process has been proposed and discussed in many studies and research projects [1][4][5][8][11][19][23][24][26][28]. This paper adopts the fundamental SLAM process to create an algorithm that can simultaneously locate landmarks, map an environment, and aid an inertial navigation system, specifically using a recursive process. This single routine will seamlessly locate and map landmarks when accurate navigation state data is provided (i.e., GPS), accurately aid the Inertial Navigation System (INS) when surveyed landmarks are available, or provide both tasks in the absence of surveyed landmarks or GPS. The theory behind a SLAM-based aiding strategy is well understood, and has proven to perform well in simulation. However, real world conditions, image processing, and sensors complicate the matter, requiring further study and design to produce a practical SLAM-based navigation system. This chapter explores the mathematics behind the problem and develops the design of a low-cost, monocular optical-inertial navigation system. The Simultaneous Location Aiding And Mapping Recursively (SLAAMR) algorithm (proposed herein) provides a robust design to accomplish the aforementioned tasks seamlessly in the airborne environment.

The SLAAMR algorithm is designed to accomplish practical INS aiding, using passive and low-cost measures, such as monocular imagery. It is specifically designed for use with a low-cost, poor quality, inertial measurement unit (IMU) aboard an airborne

platform. The use of a recursive process, specifically the extended Kalman filter (EKF), optimizes aiding action without the need of a batch estimator. This strategy reduces the hardware requirements and increases the robustness and applicability of the design

This chapter develops the error propagation model and bearings-only measurement model specific to the SLAAMR algorithm. The observability and measure of aiding action provided by bearings-only measurements are explored through the examination of the observability grammian matrix and geometric dilution of precision. This observability study identifies and proposes solutions to the weaknesses of a SLAM-based system. The proposition that INS drift can be slowed by tracking self-located/surveyed landmarks is discussed and a practical strategy is developed. The critical problem of using monocular imagery to resolve the 3-dimentions modeled in an EKF is discussed, and practical solutions are proposed. Finally, the principles of the SLAAMR algorithm design are developed, and summarized along with the algorithmic architecture. The SLAAMR algorithm differs from traditional SLAM in that it departs from theoretical conjecture and is designed to solve practical problems posed by realistic implementation limitations.

Figure 3.1 illustrates the underlying principle that allows SLAAMR to work. An aircraft using an onboard INS estimates its position, velocity, and attitude (collectively the navigation state or nav-state) as it flies, but this estimate drifts from truth. Along the way, a camera is used to track stationary landmarks on the ground. The position of these landmarks is either known or estimated. The navigation state and landmark positions are used to estimate where the landmark should appear in successive images. The

differences between actual landmark image projections and the estimates are used to resolve a 3-dimensional position error, which correlates to error in the aircraft navigation state.

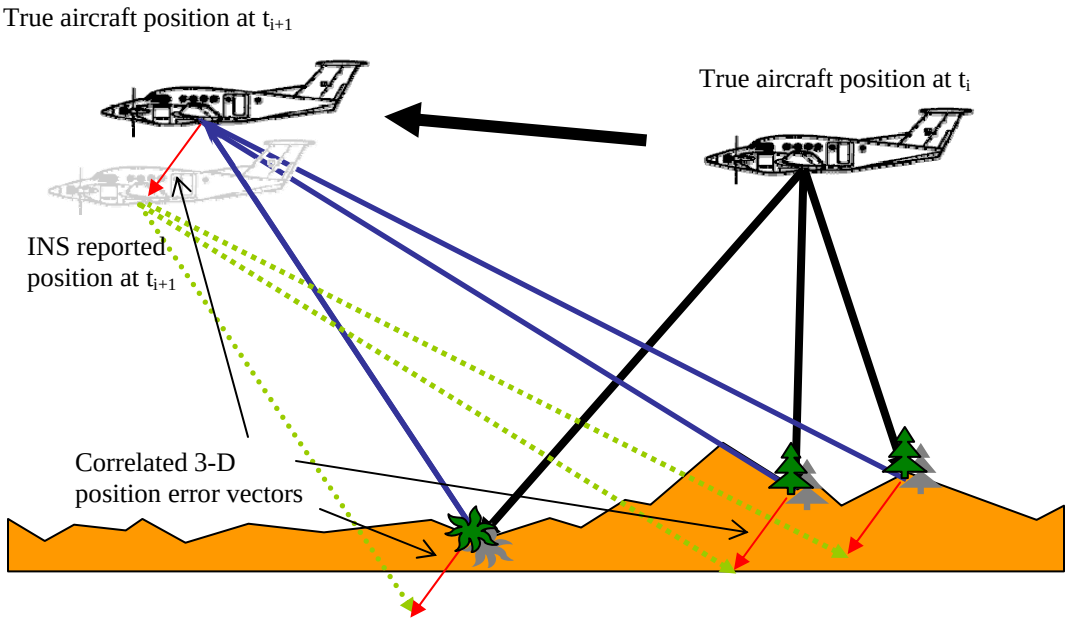

Figure 3.1: Tracking landmarks yields error vectors correlated with INS drift.

This chapter discusses the theory behind the SLAAMR algorithm and develops the case for its underpinning assumptions and tenet of design. A rigorous study of the underlying measurement equations and navigation state observability follows.

3.1 Basic EKF System Model

The underlying mechanism for SLAAMR is a low-cost inertial navigation system (INS) aided by bearings-only image (and other passive) measurements, via a feedback error state modeled by an extended Kalman filter (EKF). Chapter 2 fully developed the EKF model and process, and this chapter will develop the specific propagation equations and measurement models.

3.1.1 Error State Propagation Model. The goal of SLAAMR is to aid low-cost INS implementing a low-quality micro-machined electromechanical systems (MEMS) inertial measurement unit (IMU). This allows the assumption of a flat and non-rotating Earth approximation for the error propagation model. As derived in Chapter 2, the base error propagation model for SLAAMR is:

$$\delta\dot{\mathbf{x}}(t) = \mathbf{F}\cdot\delta\mathbf{x}(t) + \mathbf{G}\cdot\mathbf{w} \tag{3.1}$$

where

$$\mathbf{F} = \begin{bmatrix} 0_3 & \mathbf{T}_{ne} & 0_3 & 0_3 & 0_3 \\ 0_3 & 0_3 & \mathbf{C}_b^n\mathbf{f}^b\times & \mathbf{C}_b^n & 0_3 \\ 0_3 & 0_3 & 0_3 & 0_3 & -\mathbf{C}_b^n \\ 0_3 & 0_3 & 0_3 & -\frac{1}{\tau_a}\mathrm{I}_3 & 0_3 \\ 0_3 & 0_3 & 0_3 & 0_3 & -\frac{1}{\tau_b}\mathrm{I}_3 \end{bmatrix}_{15x15} \tag{3.2}$$

and

$$\mathbf{G} = \begin{bmatrix} 0_3 & 0_3 & 0_3 & 0_3 \\ \mathbf{C}_b^n & 0_3 & 0_3 & 0_3 \\ 0_3 & -\mathbf{C}_b^n & 0_3 & 0_3 \\ 0_3 & 0_3 & \mathrm{I}_3 & 0_3 \\ 0_3 & 0_3 & 0_3 & \mathrm{I}_3 \end{bmatrix}_{15x12} \tag{3.3}$$

where, $\mathbf{T}_{ne}$ transforms velocity errors from units of m/s to geodetic equivalents, $\mathbf{f}^b\times$ is the skew symmetric form of the specific force vector resolved in the *b-frame*, $\mathbf{C}_b^n$ is the *b-frame* to *n-frame* DCM, and $\tau_{a,b}$ are the accelerometer and rate gyro bias time constants.

The 15-state error vector ($\delta\mathbf{x}(t)$) is also defined in Chapter 2:

$$\delta\mathbf{x} = \begin{bmatrix} \delta\mathbf{p} & \delta\mathbf{v}_{eb}^n & \mathbf{e}_{nb}^n & \mathbf{a}^b & \mathbf{b}^b \end{bmatrix}^{\mathsf{T}} \tag{3.4}$$

This basic navigation error state vector will be augmented to include landmark position estimates in Section 3.2.4. The dynamics driving noise associated with the inertial sensors and their bias' ($\mathbf{w}$) is defined:

$$\mathbf{w} = \begin{bmatrix} \mathbf{R}_a & 0_3 & 0_3 & 0_3 \\ 0_3 & \mathbf{R}_b & 0_3 & 0_3 \\ 0_3 & 0_3 & \mathbf{Q}_a & 0_3 \\ 0_3 & 0_3 & 0_3 & \mathbf{Q}_b \end{bmatrix}_{12x12} \tag{3.5}$$

where $\mathbf{R}_a$ and $\mathbf{R}_b$ are the 3x3 noise strength (intensity) matrices for accelerometer and rate gyro (triad) measurement noise, and $\mathbf{Q}_a$ and $\mathbf{Q}_b$ are the 3x3 process driving noise strength (intensity) matrices for the accelerometer and rate gyro (triad) bias drift rates (modeled as first order Gauss-Markov processes).

3.1.2 Linearized Measurement Model. As described in Chapter 2, the SLAAMR measurement model takes the form:

$$\mathbf{z}(t_{i+1}) = \mathbf{h}[\mathbf{x}(t_{i+1}^{-}), t_{i+1}]\Big|_{\hat{\mathbf{x}}} + \mathbf{H}[t_{i+1}; \mathbf{x}(t_{i+1}^{-})]\Big|_{\hat{\mathbf{x}}} \delta\mathbf{x}(t_{i+1}) + \mathbf{v}(t_{i+1}) \tag{3.6}$$

where $\mathbf{h}[\mathbf{x}(t_{i+1}^{-}), t_{i+1}]$ is the non-linear measurement equation, and $\mathbf{H}[t_{i+1}; \mathbf{x}(t_{i+1}^{-})]$ is the matrix partial derivative of $\mathbf{h}$. $\mathbf{z}(t_{i+1})$ is the feature position measurement expressed in the image frame, and $\mathbf{v}$ is the measurement noise driven by system optics, etc. Again, $\delta\mathbf{x}(t_i)$ will be augmented to include landmark position estimates (discussed in Section 3.2.4), but the form of Equation (3.6) remains the same. The following sections develop this measurement model for bearings-only image measurements.

3.1.3 Initial Landmark Position Estimation. Before an EKF can begin, it requires an initial estimate of the state variables and associated covariances. In the feature tracking case, the state variables are the navigation and tracked object position error states. Additionally, the EKF requires an estimated nominal trajectory in order to perform the linearization. This trajectory includes navigation and tracked landmark positions. Initial estimates of the navigation whole and error states are obtained from

traditional, well-established INS alignment/transfer alignment techniques, and will not be discussed here [23]. Initial object position estimates must be derived from one of two categories of techniques: third party location (surveyed), and auto-location (self-surveyed). The initial position estimate provides a nominal trajectory about which the EKF error model is linearized. The initial error in this position is assumed to be a zero-mean and Gaussian (with an associated covariance matrix). Subsequent iterations of the EKF refine the initial position and error covariance estimates.

3.1.3.1 Third Party Survey. When a third party source is used to locate a landmark, any errors in the estimated landmark position are independent of the EKF navigation state errors. Thus, the uncertainty of the position estimate is strictly a function of the third party system. Determining an object's location through satellite imagery or a GPS site survey are examples of this method. Because the aircraft and its sensors play no part in this process, the errors associated with third party survey are completely independent of the navigation state and measurement equation. In the case of differential GPS position estimates, the uncertainty covariance can be sub-meter and elliptical. These types of landmarks will henceforth be known as surveyed landmarks.

3.1.3.2 Self-Survey or Auto-location. Self-survey or auto-location involves determining the position of a landmark of opportunity (LOO) through measurements taken from onboard sensors, in this case a camera. The camera measures the relative position of the landmark to the aircraft. This relative position is then added to the aircraft position. This technique has been demonstrated through the use of binocular vision, multiple image batch estimation, or active sensors (such as an EO/IR targeting pod) [26].

Single image monocular techniques are the primary method explored herein, because monocular imagery is a more practical and low-cost option for aircraft. Whichever technique is used, the measurement equation for auto-location estimates take the form:

$$\hat{\mathbf{p}}_{tgt} = \Delta_{tgt} + \mathbf{p}_{\mathbf{INS}} \tag{3.7}$$

where the initial landmark position estimate ($\hat{\mathbf{p}}_{tgt}$) is the sum of INS calculated aircraft position ($\mathbf{p}_{\mathbf{INS}}$) and the relative position measurement/estimate (Δ_{tgt}). The relative position measurement/estimate is a function of the whole INS-provided navigation state ($\mathbf{x}_{\mathbf{c}}$) and the sensor-produced measurement/estimate ($\bar{\mathbf{z}}$).

$$\Delta_{tgt} = \mathbf{h}_{\Delta}\left(\mathbf{x}, \mathbf{z}\right)\Big|_{\mathbf{x}_{\mathbf{c}}, \bar{\mathbf{z}}} \tag{3.8}$$

where $\mathbf{h}_{\Delta}$ is a non-linear function used to determine the landmark's relative position. Specific examples of $\mathbf{h}_{\Delta}$ are discussed in Section 3.5. The associated uncertainty of the object position estimate is derived through perturbation analysis of this equation. The error perturbations are related as:

$$\begin{aligned} \delta\hat{\mathbf{p}}_{tgt} &= \delta\Delta_{tgt} + \delta\mathbf{p} \\ &= \frac{\delta\mathbf{h}_{\Delta}\left(\mathbf{x}, \mathbf{z}\right)\Big|_{\mathbf{x}_{\mathbf{c}}, \bar{\mathbf{z}}}}{\delta\mathbf{x}}\delta\mathbf{x} + \frac{\delta\mathbf{h}_{\Delta}\left(\mathbf{x}, \mathbf{z}\right)\Big|_{\mathbf{x}_{\mathbf{c}}, \bar{\mathbf{z}}}}{\delta\mathbf{z}}\delta\mathbf{z} + \delta\mathbf{p} \end{aligned} \tag{3.9}$$

These errors are assumed to be zero mean and Gaussian. Therefore, the error covariance of the object position estimate is equal to the expected value of the outer product of the errors.

$$\begin{aligned}\mathbf{P}_{tgt} &= \mathbf{E}\left[\delta\hat{\mathbf{p}}_{tgt}\delta\hat{\mathbf{p}}_{tgt}^{\mathsf{T}}\right] \\ &= \mathbf{E}\left[\left(\delta\Delta_{tgt}+\delta\mathbf{p}\right)\left(\delta\Delta_{tgt}+\delta\mathbf{p}\right)^{\mathsf{T}}\right] \\ &= \mathbf{E}\left[\delta\Delta_{tgt}\delta\Delta_{tgt}^{\mathsf{T}}\right]+\mathbf{E}\left[\delta\Delta_{tgt}\delta\mathbf{p}^{\mathsf{T}}\right]+\mathbf{E}\left[\delta\mathbf{p}\delta\Delta_{tgt}^{\mathsf{T}}\right]+\mathbf{E}\left[\delta\mathbf{p}\delta\mathbf{p}^{\mathsf{T}}\right]\end{aligned} \tag{3.10}$$

The aircraft position errors are independent of the relative position estimate, but in practice, the measurement equation is evaluated about the nominal, errant navigation state. The assumption is made that nominal is *close* to truth. Thus, the aircraft position errors are approximated to be independent of the relative position estimate. The covariance expression simplifies to:

$$\begin{aligned}\mathbf{P}_{tgt} &= \mathbf{E}\left[\delta\Delta_{tgt}\delta\Delta_{tgt}^{\mathsf{T}}\right]+\mathbf{E}\left[\delta\mathbf{p}\delta\mathbf{p}^{\mathsf{T}}\right] \\ &= \mathbf{E}\left[\left(\frac{\delta\mathbf{h}_{\Delta}}{\delta\mathbf{x}}\delta\mathbf{x}+\frac{\delta\mathbf{h}_{\Delta}}{\delta\mathbf{z}}\delta\mathbf{z}\right)\left(\frac{\delta\mathbf{h}_{\Delta}}{\delta\mathbf{x}}\delta\mathbf{x}+\frac{\delta\mathbf{h}_{\Delta}}{\delta\mathbf{z}}\delta\mathbf{z}\right)^{\mathsf{T}}\right]+\mathbf{E}\left[\delta\mathbf{p}\delta\mathbf{p}^{\mathsf{T}}\right] \\ &= \mathbf{E}\left[\frac{\delta\mathbf{h}_{\Delta}}{\delta\mathbf{x}}\delta\mathbf{x}\delta\mathbf{x}^{\mathsf{T}}\frac{\delta\mathbf{h}_{\Delta}}{\delta\mathbf{x}}^{\mathsf{T}}\right]+\mathbf{E}\left[\frac{\delta\mathbf{h}_{\Delta}}{\delta\mathbf{x}}\delta\mathbf{x}\delta\mathbf{z}^{\mathsf{T}}\frac{\delta\mathbf{h}_{\Delta}}{\delta\mathbf{z}}^{\mathsf{T}}\right] \\ &+\mathbf{E}\left[\frac{\delta\mathbf{h}_{\Delta}}{\delta\mathbf{z}}\delta\mathbf{z}\delta\mathbf{x}^{\mathsf{T}}\frac{\delta\mathbf{h}_{\Delta}}{\delta\mathbf{x}}^{\mathsf{T}}\right]+\mathbf{E}\left[\frac{\delta\mathbf{h}_{\Delta}}{\delta\mathbf{z}}\delta\mathbf{z}\delta\mathbf{z}^{\mathsf{T}}\frac{\delta\mathbf{h}_{\Delta}}{\delta\mathbf{z}}^{\mathsf{T}}\right]+\mathbf{E}\left[\delta\mathbf{p}\delta\mathbf{p}^{\mathsf{T}}\right] \\ &= \mathbf{H}_{\mathbf{x}}\mathbf{P}_{\mathbf{xx}}\mathbf{H}_{\mathbf{x}}^{\mathsf{T}}+\mathbf{H}_{\mathbf{x}}\mathbf{P}_{\mathbf{xz}}\mathbf{H}_{\mathbf{z}}^{\mathsf{T}}+\mathbf{H}_{\mathbf{z}}\mathbf{P}_{\mathbf{zx}}\mathbf{H}_{\mathbf{x}}^{\mathsf{T}}+\mathbf{H}_{\mathbf{z}}\mathbf{P}_{\mathbf{zz}}\mathbf{H}_{\mathbf{z}}^{\mathsf{T}}+\mathbf{P}_{\mathbf{pp}}\end{aligned} \tag{3.11}$$

where $\mathbf{P_{pp}}$, $\mathbf{P_{xx}}$ and $\mathbf{P_{zz}}$ are the aircraft position, navigation state, and measurement error covariances, respectively. $\mathbf{P_{xz}}$ and $\mathbf{P_{zx}}$ are the cross correlation matrices between the navigation state and the measurements. $\mathbf{H_x}$ and $\mathbf{H_z}$ are the partial derivatives of $\mathbf{h}_\Delta(\mathbf{x},\mathbf{z})$ with respect to $\mathbf{x}$ and $\mathbf{z}$, evaluated at $\mathbf{x_c}$ and $\bar{\mathbf{z}}$. The camera optical uncertainties are independent of the navigation state errors, allowing the simplification:

$$\mathbf{P}_{tgt} = \mathbf{H_x P_{xx} H_x}^{\mathbf{T}} + \mathbf{H_z P_{zz} H_z}^{\mathbf{T}} + \mathbf{P_{pp}} \tag{3.12}$$

In the case in which a perfect relative measurement can be taken ($\delta\Delta_{tgt}$=0), the uncertainty in the object's initial position estimate can be no better that the uncertainty in the aircraft's own position estimate. Precise optics and continued bearings measurements could theoretically refine Δ_{tgt} to a near perfect estimate, but since this is a relative estimate, the uncertainty contribution of the aircraft's own position will always remain. Thus, the lower limit to LOO position estimate certainty is driven by $\mathbf{P_{pp}}$. A tracked LOO position estimate uncertainty can approach, but never exceed that of the host aircraft. When considering surveyed landmarks, the reverse is true; the aircraft's position uncertainty can approach, but never exceed that of the best surveyed landmark.

3.2 Feature Tracking Measurement Model

Tracking features in successive images is the underpinning process that will allow the SLAAMR algorithm to operate. It involves multiple steps from generating features, to matching them correctly, and eventually mapping their measurements into the navigation state. For a landmark projected onto an image to be tracked, two things must occur: the associated feature must precisely, accurately correspond to the same geodetic position in each frame, and it must be correctly and consistently identified.

3.2.1 Feature Generation. In his work, Veth implemented a scale-invariant feature transform (SIFT©) created by Lowe [26][11]. The SIFT© generates 'key points' for distinct features in an image. Each key point contains a camera frame position (in pixel coordinates), scale, orientation and a distinct descriptor allowing matching to occur. Veth's work demonstrated that SIFT© is a practical and functional feature generator and was used in the research herein. Figure 3.2 shows an image from flight test with overlaid features generated by SIFT©. Feature scale, orientation and descriptor data are not shown.

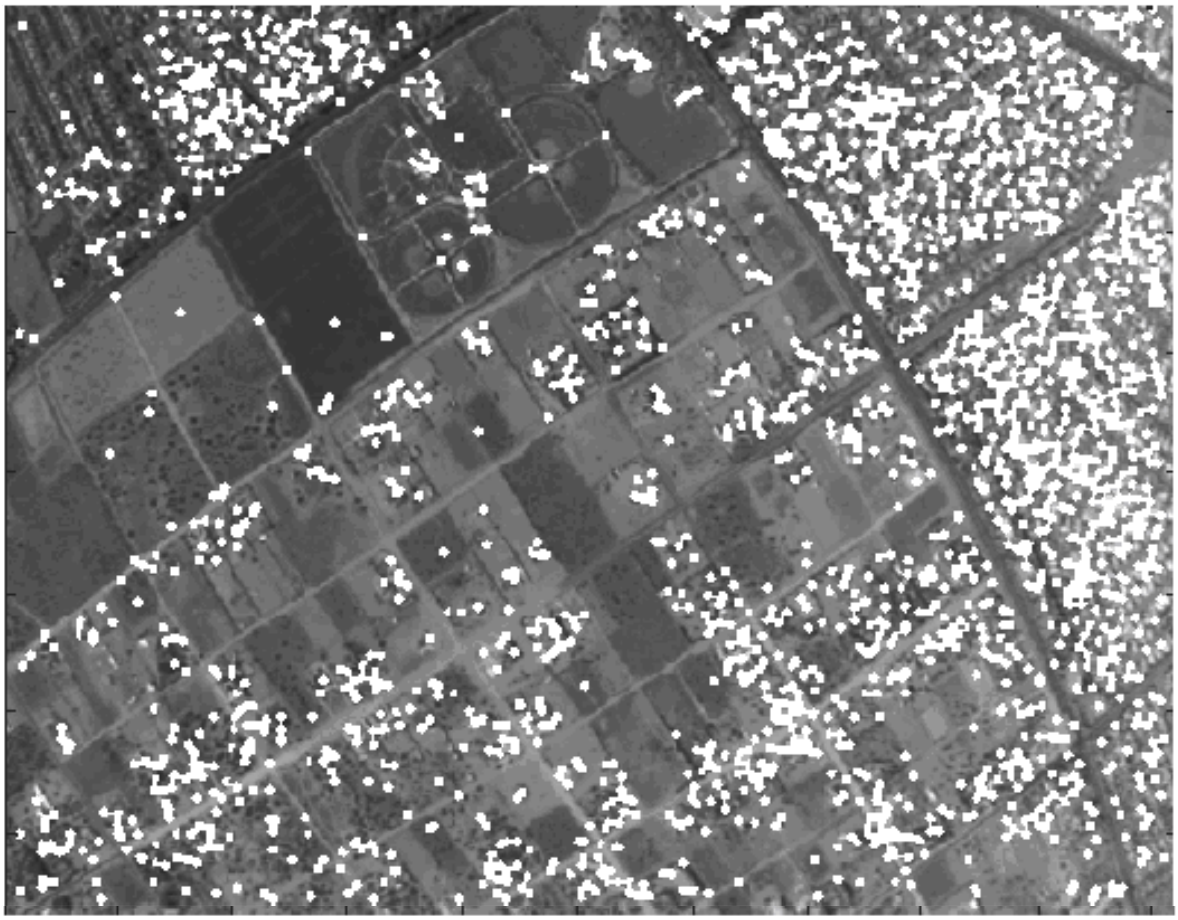

Figure 3.2: Flight test image with overlaid SIFT© generated features. The features are identified by a dot.

3.2.2 Feature Matching. The descriptor generated by SIFT© is a distinct 128-element vector. Figure 3.3 plots the 128-descriptor values versus the element's location (for one specific feature), compares it to a matched feature in another image and a feature that does not match.

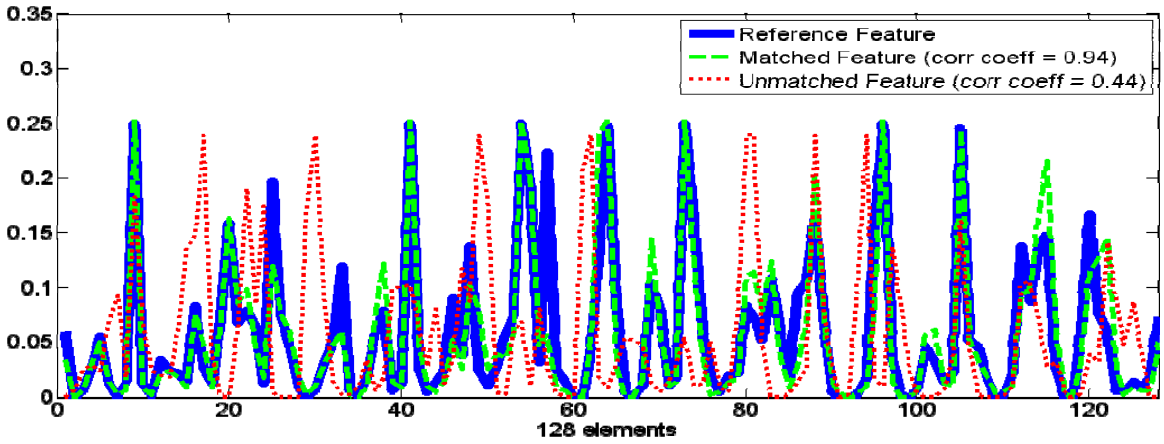

Figure 3.3: Representative descriptor values versus their element location.

The nature of the descriptor allows accurate matching using an auto-correlation routine. A correlation coefficient (ρ) is derived for a frame-to-frame feature comparison via the dot product of the two feature descriptor vectors, normalized by the dot product of the first feature descriptor with itself. The correlation routine implemented is

$$\rho = \mathrm{d}_a \bullet \mathrm{d}_b^{\mathsf{T}} \Big/ \mathrm{d}_a \bullet \mathrm{d}_a^{\mathsf{T}} \tag{3.13}$$

where d_a and d_b are the descriptor vectors of the two features being compared. The coefficient ρ varies from 1 (perfect match) to potentially -1. Practically, the range is 0.5-1.0 because of the generally positive descriptor element values. A match is declared if ρ exceeds a threshold derived for a good match.

The use of SIFT© allows the SLAAMR algorithm to operate in nearly any environment, and travel long distances without the need to revisit old landmarks.

Previous work on SLAM algorithms used a set of pre-placed targets in a confined area [5]. The targets were found and matched using a template matching algorithm. This approach worked well, but limits environment in which it can operate and relies on a priori information about the targets (appearance) despite not knowing their location.

For any given image, SIFT© produces hundreds to thousands of features. Applying the correlation equation to every feature (in order to find a match) is computationally impractical and could lead to potential false matches. The EKF naturally predicts the geodetic position of all tracked landmarks and assigns an uncertainty volume to that prediction. Projecting this estimated position and uncertainty in subsequent images allows the SLAAMR algorithm to spatially constrain which features will be considered for a match. The 3-dimensional landmark position and uncertainty project into the image as a 2-dimensional pixel location and corresponding ellipsoid. Figure 3.4 illustrates this principle.

The cross and ellipse represent the estimated feature projection and 3-sigma uncertainty bound (3 standard deviations of ~99% confidence). Only features that fall within the ellipse are considered for a match. If none exceed the threshold for a good match, the track is considered broken.

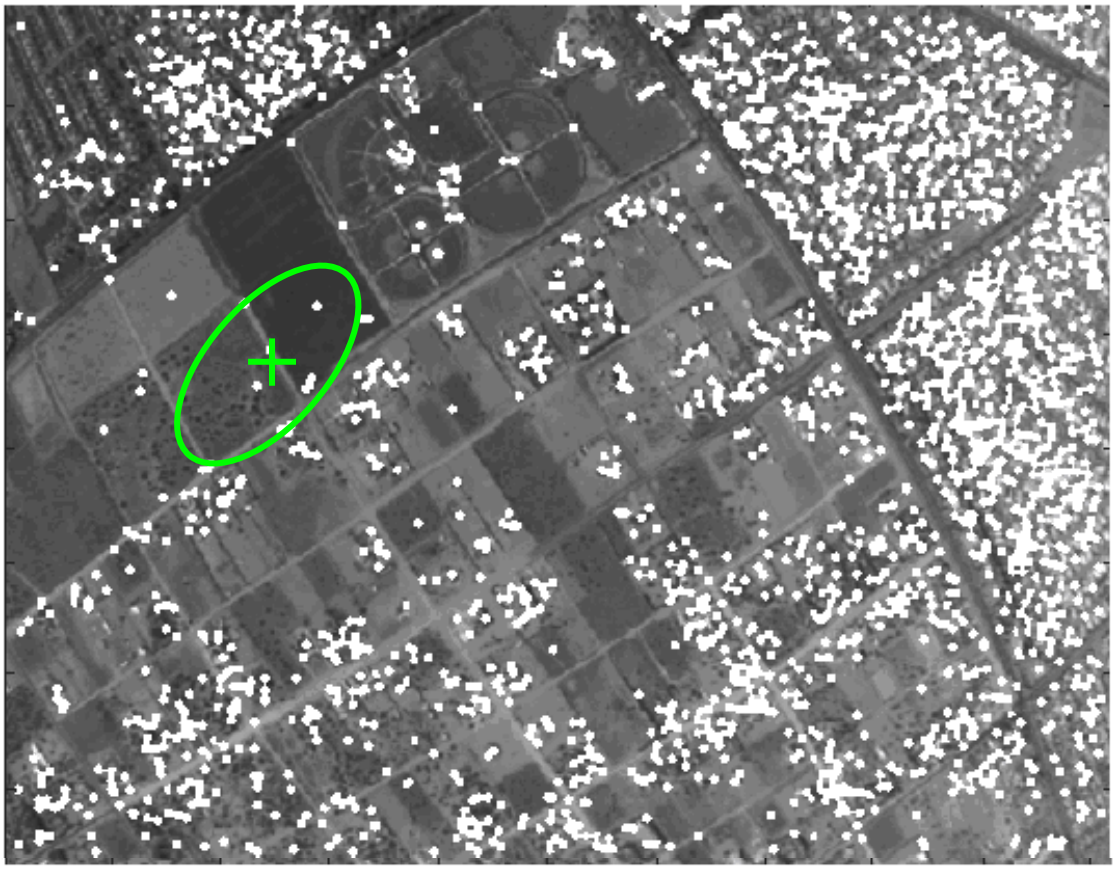

Figure 3.4: Constrained feature search volume.

The projection of landmark position and uncertainty into the image is accomplished via the measurement vector ($\mathbf{h}[\mathbf{x}(t_i^-), t_i]$) and observation matrix ($\mathbf{H}[t_i; \mathbf{x}(t_i^-)]$), which are developed in the next section. The general form of the estimated landmark projection is

$$\hat{\mathbf{z}}(t_i) = \mathbf{h}[\mathbf{x}_{\mathbf{INS}}(t_i)] \tag{3.14}$$

where $\hat{\mathbf{z}}(t_{i+1})$ is the estimated projection expressed in pixel coordinates, and $\mathbf{x}_{\mathbf{INS}}(t_{i+1})$ is the INS derived navigation state. The uncertainty in the estimated projection is mapped into the image by:

$$\mathbf{P}_{zz} = \mathbf{H}[\mathbf{x}_{\mathbf{INS}}(\mathrm{t_i})] \cdot \mathbf{P}_{xx} \cdot \mathbf{H}[\mathbf{x}_{\mathbf{INS}}(\mathrm{t_i})]^{\mathbf{T}} + \mathbf{R} \tag{3.15}$$

where $\mathbf{P}_{zz}$ is a 2x2 covariance matrix describing the 2-dimentional uncertainty ellipse, $\mathbf{P}_{\mathbf{xx}}$ is error state uncertainty covariance matrix and $\mathbf{R}$ is the measurement noise covariance matrix.

3.2.3 Image-Only Measurement Equation. As described in Chapter 2, the position vector of a tracked landmark, resolved in the *c-frame* ($\mathbf{s}^c$), is a function of aircraft and landmark geodetic position ($\mathbf{p}$ and $\mathbf{p}_{tgt}$), camera mounting position and alignment (relative to the IMU, $\mathbf{p}^b_{cam}$ and $\mathbf{C}^b_c$), and aircraft navigation state derived transformations ($\mathbf{T}_{en}$ and $\mathbf{C}^n_b$), where $\mathbf{C}^c_n = \mathbf{C}^c_b \mathbf{C}^b_n$:

$$\mathbf{s}^c = \mathbf{C}^c_n \left[\mathbf{T}_{en} \left(\mathbf{p}_{tgt} - \mathbf{p} \right) - \mathbf{C}^n_b \mathbf{p}^b_{cam} \right] \tag{3.16}$$

where $\mathbf{T}_{en}$ transforms the relative difference between aircraft and the tracked landmark position from geodetic units into meters. The relative position vector between the tracked object and the aircraft can be expressed as:

$$\Delta = \mathbf{T}_{en} \left(\mathbf{p}_{tgt} - \mathbf{p} \right) - \mathbf{C}^n_b \mathbf{p}^b_{cam} \tag{3.17}$$

The *c-frame* object position vector is transformed into the homogeneous image measurement expressed in camera pixel location coordinates:

$$\begin{aligned}\underline{\mathbf{z}}_p &= \frac{1}{s_z^c}\mathbf{T}_c^{pix}\mathbf{s}^c \\ &= \frac{1}{s_z^c}\mathbf{T}_c^{pix}\mathbf{C}_n^c\left[\mathbf{T}_{en}\left(\mathbf{p}_{tgt}-\mathbf{p}\right)-\mathbf{C}_b^n\mathbf{p}_{cam}^b\right] \\ &= \frac{1}{s_z^c}\mathbf{T}_c^{pix}\mathbf{C}_n^c\Delta\end{aligned} \tag{3.18}$$

If the magnitude of $\mathbf{p}_{cam}^b$ is small (as it was in this study), it can be ignored in order to simplify calculations. This step is not required, nor does the inclusion of a non-trivial $\mathbf{p}_{cam}^b$ change the problem or analysis. The scalar s_z^c is the z-component of the object position vector. It can be expressed as:

$$s_z^c = \begin{bmatrix}0 & 0 & 1\end{bmatrix}\mathbf{C}_n^c\Delta \tag{3.19}$$

The projection of the object's image onto the image plane is defined as:

$$\mathbf{s}^{proj} = \left[x_f, y_f, f\right]^{\mathsf{T}} \tag{3.20}$$

and is related to the *c-frame* object position vector ($\mathbf{s}^c$) as follows

$$\mathbf{s}^{proj} = \left(\frac{f}{s_z^c}\right)\mathbf{s}^c \tag{3.21}$$

Substituting into Equation (3.18) yields

$$\underline{\mathbf{z}}_p = \mathbf{T}_c^{pix} \frac{1}{f} \mathbf{s}^{proj} \tag{3.22}$$

Equation (3.22) is over-determined, allowing a simplification. Optical measurements have the "*strength"* of two equations only; image projections onto a focal plane are two-dimensional. $\underline{\mathbf{z}}_p$ is a homogeneous vector, and there is no need to calculate the third component (which is a constant 1). $\mathbf{s}^{proj}$ also has a known third component, the focal length. These properties simplify the measurement equation development.

The *c-frame* object position vector ($\mathbf{s}^c$) and image projection vector ($\mathbf{s}^{proj}$) form similar triangles and are scaled by the ratio of their lengths.

$$\frac{1}{\left\|\mathbf{s}^{proj}\right\|} \begin{bmatrix} x_f \\ y_f \\ f \end{bmatrix} = \frac{1}{\left\|\mathbf{s}^c\right\|} \mathbf{s}^c \tag{3.23}$$

where the length of the $\mathbf{s}^{proj}$ and $\mathbf{s}^c$ are defined by the 3-D norm

$$\left\|\mathbf{s}^{proj}\right\| = \sqrt{x_f{}^2 + y_f{}^2 + f^2} \tag{3.24}$$

Because $\mathbf{C}_n^c$ is unitary

$$\left\|\mathbf{s}^c\right\| = \left\|\Delta\right\| = \sqrt{\Delta_x{}^2 + \Delta_y{}^2 + \Delta_z{}^2} \tag{3.25}$$

Combining equations yields:

$$\begin{bmatrix} x_f \\ y_f \\ f \end{bmatrix} = \frac{\sqrt{{x_f}^2 + {y_f}^2 + f^2}}{\|\Delta\|} \mathbf{C}_n^c \Delta \tag{3.26}$$

The focal length is the third component of the vector on the left hand side of Equation (3.26), and can be expressed as

$$f = \frac{\sqrt{{x_f}^2 + {y_f}^2 + f^2}}{\|\Delta\|} \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \mathbf{C}_n^c \Delta \tag{3.27}$$

Equation (3.27) is used to eliminate the scalar on the right hand side of Equation (3.22). Isolating the unknown scalar (which also features the range to the landmark) yields:

$$\frac{\sqrt{{x_f}^2 + {y_f}^2 + f^2}}{\|\Delta\|} = \frac{f}{\begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \mathbf{C}_n^c \Delta} \tag{3.28}$$

Substituting into the original relationship yields the two measurement equations:

$$\begin{bmatrix} x_f \\ y_f \end{bmatrix} = \frac{f}{\begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \mathbf{C}_n^c \Delta} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \mathbf{C}_n^c \Delta \tag{3.29}$$

By dropping the homogeneous third component, the two-dimensional measurement equation vector ($\mathbf{z}_p$) becomes

$$\begin{aligned}\mathbf{z}_p &= \begin{bmatrix}1 & 0 & 0\\ 0 & 1 & 0\end{bmatrix}\mathbf{T}_c^{pix}\frac{1}{f}\mathbf{s}^{proj}\\ &= \begin{bmatrix}1 & 0 & 0\\ 0 & 1 & 0\end{bmatrix}\frac{\mathbf{T}_c^{pix}}{f}\begin{bmatrix}x_f\\ y_f\\ f\end{bmatrix}\end{aligned} \tag{3.30}$$

3.2.4 Augmented State Vector for Landmark Position Estimation. Augmenting the navigation state vector with the landmark's position vector allows the measurement model to fit the Kalman filter mold described in Chapter 2. It is also important to do this in order to model the errors in the knowledge of the landmark's position, for although it does not move, its true location cannot be known for certain. Thus, the augmented state vector is

$$\mathbf{x} = \begin{bmatrix}\mathbf{p}\\ \mathbf{v}_{eb}^n\\ \Psi_{nb}^n\\ \mathbf{p}_{tgt}\end{bmatrix} = \begin{bmatrix}L & l & h & \vdots & v_N & v_E & v_D & \vdots & \phi & \theta & \psi & \vdots & L_{tgt} & l_{tgt} & h_{tgt}\end{bmatrix}^{\mathsf{T}} \tag{3.31}$$

Note that the accelerometer and gyro biases are not included in the navigation state vector for this discussion. These bias states are normally included in the state vector, but play no part in optical measurements. For the sake of readability and without loss of generality, they are thus excluded from this discussion.

The associated augmented error state vector is then:

$$\delta\mathbf{x}=\begin{bmatrix}\delta\mathbf{p}\\ \delta\mathbf{v}^{n}_{eb}\\ \mathbf{e}^{n}_{nb}\\ \delta\mathbf{p}_{tgt}\end{bmatrix}=\begin{bmatrix}\delta L & \delta l & \delta h & \vdots & \delta v_N & \delta v_E & \delta v_D & \vdots & \varepsilon_x & \varepsilon_y & \varepsilon_z & \vdots & \delta L_{tgt} & \delta l_{tgt} & \delta h_{tgt}\end{bmatrix}^{\mathsf{T}} \quad (3.32)$$

3.2.5 Linearized Image-Only Measurement Equation. In order to implement an EKF, the measurement equation must be linearized. Recall the form of the non-linear measurement model [15].

$$\bar{\mathbf{z}}_p(t)=\mathbf{h}[\mathbf{x}(t)]+\mathbf{v}(t) \quad (3.33)$$

Since $\Delta,\mathbf{C}^{c}_{n}$ are derived from the navigation state $\mathbf{x}(t)$, the deterministic portion of $\bar{\mathbf{z}}_p(t)$ ($\mathbf{h}[\mathbf{x}(t)]$) can be re-written:

$$\mathbf{h}[\mathbf{x}(t)]=\begin{bmatrix}1 & 0 & 0\\ 0 & 1 & 0\end{bmatrix}\frac{\mathbf{T}^{pix}_{c}}{f}\begin{bmatrix}x_f\\ y_f\\ f\end{bmatrix} \quad (3.34)$$

$\mathbf{T}^{pix}_{c}$, and f are constant, thus a simpler measurement equation may be considered

$$\mathbf{h}'[\Delta,\mathbf{C}^{c}_{n}]=\begin{bmatrix}x_f\\ y_f\end{bmatrix} \quad (3.35)$$

where

$$\mathbf{h}[\mathbf{x}(t)] = \mathbf{h}\left(\mathbf{h}'[\Delta, \mathbf{C}_n^c]\right) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \frac{\mathbf{T}_c^{pix}}{f} \begin{bmatrix} \mathbf{h}'[\Delta, \mathbf{C}_n^c] \\ f \end{bmatrix} \tag{3.36}$$

and

$$\bar{\mathbf{z}}_p(t) = \mathbf{h}\left(\mathbf{h}'[\Delta, \mathbf{C}_n^c]\right) + \mathbf{v}(t) \tag{3.37}$$

Because $\mathbf{T}_c^{pix}$ and f are constant, the partial derivative of $\mathbf{h}[\mathbf{x}(t)]$ with respect to $\mathbf{x}(t)$ is the partial derivative of $\mathbf{h}'[\Delta, \mathbf{C}_n^c]$ with respect to $\mathbf{x}(t)$ (or equivalently, Δ and $\mathbf{C}_n^c$) multiplied by a constant matrix.

To develop the linearization of the measurement equation, unit vectors must first be introduced:

$$\mathbf{e}_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \quad \mathbf{e}_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \quad \mathbf{e}_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \tag{3.38}$$

The measurement equation can now be written more compactly:

$$\mathbf{h}'[\Delta, \mathbf{C}_n^c] = \begin{bmatrix} x_f \\ y_f \end{bmatrix} = f \begin{bmatrix} \dfrac{\mathbf{e}_1^T \mathbf{C}_n^c \Delta}{\mathbf{e}_3^T \mathbf{C}_n^c \Delta} \\ \dfrac{\mathbf{e}_2^T \mathbf{C}_n^c \Delta}{\mathbf{e}_3^T \mathbf{C}_n^c \Delta} \end{bmatrix} \tag{3.39}$$

To linearize the measurement Equation (3.39), the partials with respect to the state variables must be taken. The partials with respect to Δ equal the partials with respect to the relative camera and landmark positions. Also, for the relatively small changes in *geodetic* position over time (that a conventional aircraft undergoes in flight), the transform between geodetic and *n-frame* units of measurement ($\mathbf{T}_{en}$) is treated as a constant matrix. This simplifies the solution only somewhat. The following is the mathematical derivation for the perturbation model used to linearize the measurement model about the INS calculated trajectory.

Now, the measurement equation elements are

$$h_1'\left(\Delta,\mathbf{C}_c^n\right)=f\frac{\left(\mathbf{e}_1^{\mathsf{T}}\mathbf{C}_n^c\Delta\right)^{\mathsf{T}}}{\left(\mathbf{e}_3^{\mathsf{T}}\mathbf{C}_n^c\Delta\right)^{\mathsf{T}}}=f\frac{\Delta^{\mathsf{T}}\mathbf{C}_c^n\mathbf{e}_1}{\Delta^{\mathsf{T}}\mathbf{C}_c^n\mathbf{e}_3} \tag{3.40}$$

$$h_2'\left(\Delta,\mathbf{C}_c^n\right)=f\frac{\left(\mathbf{e}_2^{\mathsf{T}}\mathbf{C}_n^c\Delta\right)^{\mathsf{T}}}{\left(\mathbf{e}_3^{\mathsf{T}}\mathbf{C}_n^c\Delta\right)^{\mathsf{T}}}=f\frac{\Delta^{\mathsf{T}}\mathbf{C}_c^n\mathbf{e}_2}{\Delta^{\mathsf{T}}\mathbf{C}_c^n\mathbf{e}_3} \tag{3.41}$$

This form of the measurement equation facilitates the calculation of the partial derivatives. Each element (h_1' and h_2') of the measurement equation takes the form:

$$y=\frac{\mathbf{x}^{\mathsf{T}}\mathbf{a}}{\mathbf{x}^{\mathsf{T}}\mathbf{b}} \tag{3.42}$$

where $\mathbf{a}$ and $\mathbf{b}$ are constant vectors.

The perturbation in y with respect to $\delta\mathbf{x}$ is thus:

$$\delta y = \frac{\partial \mathbf{h}'}{\partial \mathbf{x}} \delta\mathbf{x} = \frac{1}{\left(\mathbf{x}^{\mathsf{T}}\mathbf{b}\right)^2} \mathbf{x}^{\mathsf{T}} \left(\mathbf{b}\mathbf{a}^{\mathsf{T}} - \mathbf{a}\mathbf{b}^{\mathsf{T}}\right) \delta\mathbf{x} \tag{3.43}$$

In Equations (3.40) and (3.41), let $\mathbf{x} = \Delta$, $\mathbf{a} = \mathbf{C}_c^n \mathbf{e}_{1,2}$, and $\mathbf{b} = \mathbf{C}_c^n \mathbf{e}_3$. This substitution yields:

$$\delta h_i' = \frac{1}{\left(\mathbf{x}^{\mathsf{T}}\mathbf{b}\right)^2} \mathbf{x}^{\mathsf{T}} \left(\mathbf{b}\mathbf{a}^{\mathsf{T}} - \mathbf{a}\mathbf{b}^{\mathsf{T}}\right) \delta\mathbf{x} = f \frac{\Delta^{\mathsf{T}} \mathbf{C}_c^n \mathbf{e}_3 \mathbf{e}_i^{\mathsf{T}} \mathbf{C}_n^c - \Delta^{\mathsf{T}} \mathbf{C}_c^n \mathbf{e}_i \mathbf{e}_3^{\mathsf{T}} \mathbf{C}_n^c}{\left(\Delta^{\mathsf{T}} \mathbf{C}_c^n \mathbf{e}_3\right)^2} \delta\Delta \tag{3.44}$$

The component due to perturbations in attitude requires further development. Consider the scalar function

$$y = \mathbf{h}'(\mathbf{x}), \quad \mathbf{h}' : \Re^l \to \Re^l \tag{3.45}$$

where the variable of interest can be written as the nonlinear relationship

$$x = \mathbf{a}^{\mathsf{T}} \mathbf{C}_c^n \mathbf{b} \tag{3.46}$$

Here, $\mathbf{a} = \Delta$ and $\mathbf{b} = \mathbf{e}_{1,2,3}$. Both are 3 dimensional column vectors. The perturbation in y caused by a perturbation in the DCM is:

$$\delta y = \frac{\partial \mathbf{h}'}{\partial \mathbf{x}} \mathbf{a}^{\mathsf{T}} \delta \mathbf{C}_c^n \mathbf{b} \tag{3.47}$$

Small changes in the DCM are driven by the small changes in the camera attitude angles' error vector $\mathbf{e}_{nc}^{n}$. The following holds true about the perturbed DCM [23]:

$$\mathbf{C}_{c\,\text{perturbed}}^{n} = \left(\mathbf{I} - \left(\mathbf{e}_{nc}^{n}\times\right)\right)\mathbf{C}_{c}^{n} = \mathbf{C}_{c}^{n} - \left(\mathbf{e}_{nc}^{n}\times\right)\mathbf{C}_{c}^{n} \simeq \mathbf{C}_{c}^{n} + \delta\mathbf{C}_{c}^{n} \tag{3.48}$$

Thus it can be declared that:

$$\delta\mathbf{C}_{c}^{n} = -\left(\mathbf{e}_{nc}^{n}\times\right)\mathbf{C}_{c}^{n} \tag{3.49}$$

Hence, the perturbation in y caused by $\mathbf{e}_{nc}^{n}$ is:

$$\delta y = -\frac{\partial \mathbf{h}'}{\partial \mathbf{x}}\Delta^{\mathsf{T}}\left(\mathbf{e}_{nc}^{n}\times\right)\mathbf{C}_{c}^{n} \tag{3.50}$$

Since the camera is rigidly connected to the *b-frame*, and $\mathbf{e}_{nc}^{n}$ are errors about the *n-frame* axis, $\mathbf{e}_{nc}^{n}$ is equivalent to the aircraft attitude angles' error vector $\mathbf{e}_{nb}^{n}$. Substituting the variables yields:

$$\delta y = f\frac{\Delta^{\mathsf{T}}\mathbf{C}_{c}^{n}\mathbf{e}_{3}\left(-\Delta^{\mathsf{T}}\left(\mathbf{e}_{nb}^{n}\times\right)\mathbf{C}_{c}^{n}\mathbf{e}_{i}\right) + \Delta^{\mathsf{T}}\mathbf{C}_{c}^{n}\mathbf{e}_{i}\left(\Delta^{\mathsf{T}}\left(\mathbf{e}_{nb}^{n}\times\right)\mathbf{C}_{c}^{n}\mathbf{e}_{3}\right)}{\left(\Delta^{\mathsf{T}}\mathbf{C}_{c}^{n}\mathbf{e}_{3}\right)^{2}} \tag{3.51}$$

Using the identity $(\mathbf{e_a}\times)\mathbf{C}_c^n\mathbf{e_b} = -\left[\left(\mathbf{C}_c^n\mathbf{e_b}\right)\times\right]\mathbf{e_a}$, $\mathbf{e}_{nb}^n$ is pulled out to the right hand side :

$$\delta\mathbf{y} = f\frac{\Delta^{\mathsf{T}}\mathbf{C}_c^n\left(\mathbf{e}_3\Delta^{\mathsf{T}}\left(\mathbf{C}_c^n\mathbf{e}_i\right)\times - \mathbf{e}_i\Delta^{\mathsf{T}}\left(\mathbf{C}_c^n\mathbf{e}_3\right)\times\right)}{\left(\Delta^{\mathsf{T}}\mathbf{C}_c^n\mathbf{e}_3\right)^2}\mathbf{e}_{nb}^n \tag{3.52}$$

where $\left(\mathbf{C}_c^n\mathbf{e}_i\right)\times$ is the skew symmetric form of the transformed vector $\mathbf{C}_c^n\mathbf{e}_i$. Combining all the perturbations for the optical measurement Equations (3.40) and (3.41) yields the perturbation model with respect to the navigation state errors:

$$\begin{aligned}\delta h_i' &= f\frac{\Delta^{\mathsf{T}}\mathbf{C}_c^n\mathbf{e}_3\mathbf{e}_i^{\mathsf{T}}\mathbf{C}_n^c - \Delta^{\mathsf{T}}\mathbf{C}_c^n\mathbf{e}_i\mathbf{e}_3^{\mathsf{T}}\mathbf{C}_n^c}{\left(\Delta^{\mathsf{T}}\mathbf{C}_c^n\mathbf{e}_3\right)^2}\delta\Delta + f\frac{\Delta^{\mathsf{T}}\mathbf{C}_c^n\left(\mathbf{e}_3\Delta^{\mathsf{T}}\left(\mathbf{C}_c^n\mathbf{e}_i\right)\times - \mathbf{e}_i\Delta^{\mathsf{T}}\left(\mathbf{C}_c^n\mathbf{e}_3\right)\times\right)}{\left(\Delta^{\mathsf{T}}\mathbf{C}_c^n\mathbf{e}_3\right)^2}\mathbf{e}_{nb}^n \\ &= \frac{f}{\left(\Delta^{\mathsf{T}}\mathbf{C}_c^n\mathbf{e}_3\right)^2}\left[\Delta^{\mathsf{T}}\mathbf{C}_c^n\left(\mathbf{e}_3\mathbf{e}_i^{\mathsf{T}} - \mathbf{e}_i\mathbf{e}_3^{\mathsf{T}}\right)\mathbf{C}_n^c\delta\Delta + \Delta^{\mathsf{T}}\mathbf{C}_c^n\left(\mathbf{e}_3\Delta^{\mathsf{T}}\left(\mathbf{C}_c^n\mathbf{e}_i\right)\times - \mathbf{e}_i\Delta^{\mathsf{T}}\left(\mathbf{C}_c^n\mathbf{e}_3\right)\times\right)\mathbf{e}_{nb}^n\right]\end{aligned} \tag{3.53}$$

To simplify, the unit vectors can be collected as such:

$$\begin{aligned}\mathbf{e}_3\mathbf{e}_1^{\,T} - \mathbf{e}_1\mathbf{e}_3^{\,T} &= -\mathbf{e}_2\times \\ \mathbf{e}_3\mathbf{e}_2^{\,T} - \mathbf{e}_2\mathbf{e}_3^{\,T} &= \mathbf{e}_1\times\end{aligned} \tag{3.54}$$

The skew symmetric form of the unit vectors are defined

$$\mathbf{E}_1 = \mathbf{e}_1\times = \begin{bmatrix}0 & 0 & 0\\ 0 & 0 & -1\\ 0 & 1 & 0\end{bmatrix},\quad \mathbf{E}_2 = \mathbf{e}_2\times = \begin{bmatrix}0 & 0 & 1\\ 0 & 0 & 0\\ -1 & 0 & 0\end{bmatrix},\quad \mathbf{E}_3 = \mathbf{e}_3\times = \begin{bmatrix}0 & -1 & 0\\ 1 & 0 & 0\\ 0 & 0 & 0\end{bmatrix} \tag{3.55}$$

Recall, the navigation error state augmented with the uncertainty in the landmark position is

$$\delta\mathbf{x} = \begin{bmatrix} \delta L & \delta l & \delta h & \vdots & \delta v_N & \delta v_E & \delta v_D & \vdots & \varepsilon_x & \varepsilon_y & \varepsilon_z & \vdots & \delta L_{tgt} & \delta l_{tgt} & \delta h_{tgt} \end{bmatrix}^{\mathsf{T}} \quad (3.56)$$

making

$$\delta\Delta = \delta\left(\mathbf{T}_{en}\left(\mathbf{p}_{tgt} - \mathbf{p}\right) - \mathbf{C}_b^n \mathbf{p}_{cam}^b\right) = \mathbf{T}_{en}\left(\delta\left(\mathbf{p}_{tgt} - \mathbf{p}\right)\right) - \delta\left(\mathbf{C}_b^n \mathbf{p}_{cam}^b\right) = \mathbf{T}_{en}\left(\delta\mathbf{p}_{tgt} - \delta\mathbf{p}\right)$$

$$\delta\Delta = \mathbf{T}_{en}\left(\begin{bmatrix} \delta L_{tgt} & \delta l_{tgt} & \delta h_{tgt} \end{bmatrix}^{\mathsf{T}} - \begin{bmatrix} \delta L & \delta l & \delta h \end{bmatrix}^{\mathsf{T}}\right) \quad (3.57)$$

Thus the simplified observation matrix is written:

$$\mathbf{H}[\mathbf{x}_c(t)] = \frac{f}{\left(\Delta^{\mathsf{T}}\mathbf{C}_c^n\mathbf{e}_3\right)^2}\begin{bmatrix} -\Delta^{\mathsf{T}}\mathbf{C}_c^n\mathbf{E}_2\mathbf{C}_n^e\mathbf{T}_{en} & \vdots & \mathbf{0}_{1\mathbf{x}3} & \vdots & \Delta^{\mathsf{T}}\mathbf{C}_c^n\left(\mathbf{e}_3\Delta^{\mathsf{T}}\left(\mathbf{C}_c^n\mathbf{e}_1\right)\times - \mathbf{e}_1\Delta^{\mathsf{T}}\left(\mathbf{C}_c^n\mathbf{e}_3\right)\times\right) & \vdots & \Delta^{\mathsf{T}}\mathbf{C}_c^n\mathbf{E}_2\mathbf{C}_n^e\mathbf{T}_{en} \\ \Delta^{\mathsf{T}}\mathbf{C}_c^n\mathbf{E}_1\mathbf{C}_n^e\mathbf{T}_{en} & \vdots & \mathbf{0}_{1\mathbf{x}3} & \vdots & \Delta^{\mathsf{T}}\mathbf{C}_c^n\left(\mathbf{e}_3\Delta^{\mathsf{T}}\left(\mathbf{C}_c^n\mathbf{e}_2\right)\times - \mathbf{e}_2\Delta^{\mathsf{T}}\left(\mathbf{C}_c^n\mathbf{e}_3\right)\times\right) & \vdots & -\Delta^{\mathsf{T}}\mathbf{C}_c^n\mathbf{E}_1\mathbf{C}_n^e\mathbf{T}_{en} \end{bmatrix} \quad (3.58)$$

The complete observation matrix is obtained by applying Equation (3.36):

$$\mathbf{H}[\mathbf{x}_c(t)] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}\frac{\mathbf{T}_c^{pix}}{f}\begin{bmatrix} \mathbf{H}'[\mathbf{x}_c(t)] \\ 0 \end{bmatrix} \quad (3.59)$$

The transform $\mathbf{T}_{en}$ merely changes the units from geodetic (radians of longitude and latitude) to meters, and is considered essentially constant. Without loss of generality, it can be pulled out of Equation (3.59), and applied to the error states before the Kalman filter update cycle. Also, a reference frame can be declared such that $\mathbf{T}_{en}$ is identity. The same can be said for the constants in front. For the sake of readability, $\mathbf{T}_{en}$ and the constants in front will be omitted from the observation matrix from this point on.

3.2.6 Tracking Many Unknown Landmarks. Tracking of multiple landmarks with position that is unknown simply requires augmenting the error state (and associated state vector) with additional landmark position states:

$$\mathbf{x} = \begin{bmatrix} \mathbf{p} \\ \mathbf{v}_{eb}^{n} \\ \boldsymbol{\Psi}_{nb}^{n} \\ \cdots \\ \mathbf{p}_{tgt1} \\ \vdots \\ \mathbf{p}_{tgt\mathbf{N}} \end{bmatrix}, \qquad \delta\mathbf{x} = \begin{bmatrix} \delta\mathbf{p} \\ \delta\mathbf{v}_{eb}^{n} \\ \mathbf{e}_{nb}^{n} \\ \cdots \\ \delta\mathbf{p}_{tgt1} \\ \vdots \\ \delta\mathbf{p}_{tgt\mathbf{N}} \end{bmatrix} \tag{3.60}$$

where $\mathbf{p}$, $\mathbf{v}$ and $\boldsymbol{\Psi}$ are the states associated with the INS, and $\mathbf{p}_{tgt\mathbf{n}}$ is the n^{th} landmark position vector.

The error propagation equations are augmented to model the stationary objects as follows:

$$\delta\dot{\mathbf{x}} = \begin{bmatrix} \mathbf{F}_{MxM} & \mathbf{0}_{Mx3} & \cdots & \mathbf{0}_{Mx3} \\ \mathbf{0}_{3xM} & \mathbf{0}_{3x3} & \cdots & \mathbf{0}_{3x3} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0}_{3xM} & \mathbf{0}_{3x3} & \cdots & \mathbf{0}_{3x3} \end{bmatrix} \delta\mathbf{x} + \begin{bmatrix} \mathbf{G}_{Mx6} \\ \mathbf{0}_{3x6} \\ \vdots \\ \mathbf{0}_{3x6} \end{bmatrix} \delta\mathbf{w} \tag{3.61}$$

where $\mathbf{F}$ and $\mathbf{G}$ are the INS (*MxM*) systems dynamics matrix and the (*Mx6*) noise transformation matrix from Chapter 2. One column and row of zeros is added for every new object position state variable added.

The measurement Equations (3.40) and (3.41) are augmented as well to reflect the added measurements and state variables. The observation matrix ($\mathbf{H}[\mathbf{x_c}(t)]$) is augmented with two additional rows for each additional tracked landmark. The i^{th} pair of rows, corresponding to the i^{th} of N tracked landmarks, are defined:

$$\mathbf{H}_i\left(\mathbf{x_c}(t)\right) = \frac{f}{\left(\Delta_i^T \mathbf{C}_c^n \mathbf{e}_3\right)^2} \begin{bmatrix} -\Delta_i^T \mathbf{C}_c^n \mathbf{E}_2 \mathbf{C}_n^c & \vdots & \mathbf{0}_{1\mathbf{x}3} & \vdots & \Delta_i^T \mathbf{C}_c^n \left(\mathbf{e}_3 \Delta_i^T \left(\mathbf{C}_c^n \mathbf{e}_1\right)\times - \mathbf{e}_1 \Delta_i^T \left(\mathbf{C}_c^n \mathbf{e}_3\right)\times\right) \\ +\Delta_i^T \mathbf{C}_c^n \mathbf{E}_1 \mathbf{C}_n^c & \vdots & \mathbf{0}_{1\mathbf{x}3} & \vdots & \Delta_i^T \mathbf{C}_c^n \left(\mathbf{e}_3 \Delta_i^T \left(\mathbf{C}_c^n \mathbf{e}_2\right)\times - \mathbf{e}_2 \Delta_i^T \left(\mathbf{C}_c^n \mathbf{e}_3\right)\times\right) \\ & \vdots & \mathbf{0}_{1\mathbf{x}(3\bullet(i-1))} & \vdots & +\Delta_i^T \mathbf{C}_c^n \mathbf{E}_2 \mathbf{C}_n^c & \vdots & \mathbf{0}_{1\mathbf{x}(3\bullet(N-i))} \\ & \vdots & \mathbf{0}_{1\mathbf{x}(3\bullet(i-1))} & \vdots & -\Delta_i^T \mathbf{C}_c^n \mathbf{E}_1 \mathbf{C}_n^c & \vdots & \mathbf{0}_{1\mathbf{x}(3\bullet(N-i))} \end{bmatrix}$$

$$\Delta_i = \mathbf{T}_{en}\left(\mathbf{p}_{tgti} - \mathbf{p}\right) - \mathbf{C}_b^n \mathbf{p}_{cam}^b \quad , \qquad i = 1, \ldots, N \tag{3.62}$$

3.2.7 Tracking Known Landmarks. If the landmark position is *known perfectly*, the state variables concerning the object location can be dropped, and substituted for constants throughout the model. *True* landmark position knowledge is reasonably attainable through an a priori survey, but will not be discussed. In the case in which a single object is tracked, the object line-of-sight (LOS) vector can be substituted for the position vector.

$$\mathbf{p} \leftarrow \Delta = \begin{bmatrix} x - x_p \\ y - y_p \\ z - z_p \end{bmatrix} \tag{3.63}$$

Thus the new navigation state error vector is now:

$$\delta\mathbf{x} = \begin{bmatrix} \delta\Delta^{\mathsf{T}} & \delta\mathbf{v}^{\mathsf{T}} & \mathbf{e}_{nb}^{n\,\mathsf{T}} \end{bmatrix}^{\mathsf{T}} \tag{3.64}$$

Because the LOS vector is now the difference between the camera position and a constant landmark position, the position and velocity derivatives, and therefore the error propagation equations, do not change from before:

$$\begin{bmatrix} \delta\dot{\Delta} \\ \delta\dot{\mathbf{v}} \\ \dot{\mathbf{e}}_{nb}^{n} \end{bmatrix} = \mathbf{F}_{9x9} \begin{bmatrix} \delta\Delta \\ \delta\mathbf{v} \\ \mathbf{e}_{nb}^{n} \end{bmatrix} + \mathbf{G}_{9x6}\delta\mathbf{w} \tag{3.65}$$

The measurement equation also remains as described before.

The case where many known landmarks are tracked is a simple extension. Since all the landmarks are known and stationary, the position of one can be related to another by differencing their positions.

$$\mathbf{p}_{tgt1} = \mathbf{p}_{tgti} + (\mathbf{p}_{tgt1} - \mathbf{p}_{tgti}), \qquad i = 2, \ldots, N \tag{3.66}$$

Because the landmarks are stationary, the differences in their positions are constant. Therefore, no matter how many known landmarks are tracked, the state dimension remains the same. The LOS vectors Δ_i used in the additional measurement equations are therefore expressed:

$$\Delta_i = \mathbf{p} - \mathbf{p}_{tgt\,i}, \qquad i = 1, \ldots, N \tag{3.67}$$

The augmented measurement equation ($\mathbf{h}''[\mathbf{x}(t)]$) simply becomes a vector of N functions taking the form of Equation (3.36).

$$\mathbf{h}''[\mathbf{x}(t)] = \begin{bmatrix} \mathbf{h}_1[\mathbf{x}(t)] \\ \vdots \\ \mathbf{h}_N[\mathbf{x}(t)] \end{bmatrix} \tag{3.68}$$

The Observation matrix then contains N pairs of rows, where the i^{th} pair of rows are:

$$\mathbf{H}_i\left(\mathbf{x}_c(t)\right)=\frac{f}{\left(\Delta_i^T\mathbf{C}_c^n\mathbf{e}_3\right)^2}\begin{bmatrix} -\Delta_i^T\mathbf{C}_c^n\mathbf{E}_2\mathbf{C}_n^c & \vdots & \mathbf{0}_{1\mathbf{x}3} & \vdots & \Delta_i^T\mathbf{C}_c^n\left(\mathbf{e}_3\Delta_i^T\left(\mathbf{C}_c^n\mathbf{e}_1\right)\times-\mathbf{e}_1\Delta_i^T\left(\mathbf{C}_c^n\mathbf{e}_3\right)\times\right) \\ \Delta_i^T\mathbf{C}_c^n\mathbf{E}_1\mathbf{C}_n^c & \vdots & \mathbf{0}_{1\mathbf{x}3} & \vdots & \Delta_i^T\mathbf{C}_c^n\left(\mathbf{e}_3\Delta_i^T\left(\mathbf{C}_c^n\mathbf{e}_2\right)\times-\mathbf{e}_2\Delta_i^T\left(\mathbf{C}_c^n\mathbf{e}_3\right)\times\right) \end{bmatrix} \tag{3.69}$$

3.3 Image-Only Measurement Observability

The strength of aiding action provided to the INS from optical measurements depends on the degree of observability of the INS error states as provided by the measurement equation. The observation matrix $\mathbf{H}\left(\mathbf{x}_c(t)\right)$ is time-dependent. Hence, the degree of observability provided by bearings-only measurements requires the calculation of the observability grammian

$$\mathbf{W}(t)=\int_0^t e^{\mathbf{F}^T t}\cdot\mathbf{H}^T\left(\mathbf{x}_c(t)\right)\mathbf{H}\left(\mathbf{x}_c(t)\right)e^{\mathbf{F}t}dt \tag{3.70}$$

Note that the observability grammian is also trajectory-dependent. A study of the observability of bearings-only measurements for a SLAM based system has been accomplished [30], but this section fully develops the problem for an airborne application in an attempt to design an aiding strategy more effectively. It is important to note that this study applies to the update cycle of the EKF and speaks nothing to the error growth

that occurs between measurements. Error growth that occurs in the EKF propagation cycle is a function of the IMU being used. This error is constrained most effectively when the measurement equation has good observability. This study also examines the continuous-measurement case, but the findings can be extended into the discrete, still-camera realm in good faith.

3.3.1 Surveyed Landmark Tracking Observability. It is useful to begin with the simpler condition of tracking landmarks with perfectly known positions (nav-state not augmented for landmark position estimation). Consider wings-level, constant-altitude flight. A downward looking camera is employed from the aircraft flying at a constant velocity (v) and constant altitude (h) over flat terrain for time (t). A landmark directly along the flight path is tracked. The INS aiding action may be thoroughly examined in this simple scenario.

The camera, body, and nav- frames are all nominally aligned. Thus, $\mathbf{C}_b^n = \mathrm{I}$. In the case where one known landmark is tracked, the 9x9 observability grammian for level, un-accelerated flight is

$$\mathbf{W}(t) = t\frac{f^2}{h^2}\begin{bmatrix}
1 & 0 & -\frac{v}{h}\left(\frac{t}{2}\right)-\frac{\Delta_x}{h} & \frac{t}{2} & 0 \\
0 & 1 & 0 & 0 & \frac{t}{2} \\
-\frac{v}{h}\left(\frac{t}{2}\right)-\frac{\Delta_x}{h} & 0 & \frac{v^2}{h^2}\left(\frac{t^2}{3}\right)+\frac{v}{h}\left(\frac{\Delta_x t}{h}\right)+\frac{\Delta_x^2}{h^2} & -\frac{v}{h}\left(\frac{t^2}{3}\right)-\frac{\Delta_x t}{2h} & 0 \\
\frac{t}{2} & 0 & -\frac{v}{h}\left(\frac{t^2}{3}\right)-\frac{\Delta_x t}{2h} & \frac{t^2}{3} & 0 \\
0 & \frac{t}{2} & 0 & 0 & \frac{t^2}{3} \\
-\frac{v}{h}\left(\frac{t^2}{3}\right)-\frac{\Delta_x t}{2h} & 0 & \frac{v^2}{h^2}\left(\frac{t^3}{4}\right)+\frac{v}{h}\left(\frac{8\Delta_x t^2}{12h}\right)+\frac{6\Delta_x^2 t}{12h^2} & -\frac{v}{h}\left(\frac{t^3}{4}\right)-\frac{\Delta_x t^2}{3h} & 0 \\
0 & \frac{gt^2}{6}-h & 0 & 0 & \frac{gt^3}{8}-\frac{ht}{2} \\
\frac{v}{h}\left(\frac{t^2 v}{3}+\frac{\Delta_x t}{1}\right)+\frac{\Delta_x^2+h^2}{h}-\frac{gt^2}{6} & 0 & \frac{v}{h}\left(\frac{gt^3}{8}-\frac{3\Delta_x^2 t}{2h}\right)-\frac{v^2}{h^2}\left(\Delta_x t^2+\frac{vt^3}{4}\right) & \frac{v}{h}\left(\frac{vt^3}{4}+\frac{2\Delta_x t^2}{3}\right)-\frac{gt^3}{8}+\frac{\Delta_x^2 t}{2h}+\frac{ht}{2} & 0 \\
 & & +\frac{\Delta_x gt^2}{6h}-\frac{vt}{2}-\frac{\Delta_x^3}{h^2}-\Delta_x & & \\
0 & \frac{vt}{2}+\Delta_x & 0 & 0 & \frac{vt^2}{3}+\frac{\Delta_x t}{2}
\end{bmatrix}$$

$$\begin{bmatrix}
-\frac{v}{h}\left(\frac{t^2}{3}\right)-\frac{\Delta_x t}{2h} & 0 & \frac{v}{h}\left(\frac{t^2 v}{3}+\frac{\Delta_x t}{1}\right)+\frac{\Delta_x^2+h^2}{h}-\frac{gt^2}{6} & 0 \\
0 & \frac{gt^2}{6}-h & 0 & \frac{vt}{2}+\Delta_x \\
\frac{v^2}{h^2}\left(\frac{t^3}{4}\right)+\frac{v}{h}\left(\frac{8\Delta_x t^2}{12h}\right)+\frac{6\Delta_x^2 t}{12h^2} & 0 & \frac{v}{h}\left(\frac{gt^3}{8}-\frac{3\Delta_x^2 t}{2h}\right)-\frac{v^2}{h^2}\left(\Delta_x t^2+\frac{vt^3}{4}\right) & 0 \\
 & & +\frac{\Delta_x gt^2}{6h}-\frac{vt}{2}-\frac{\Delta_x^3}{h^2}-\Delta_x & \\
-\frac{v}{h}\left(\frac{t^3}{4}\right)-\frac{\Delta_x t^2}{3h} & 0 & \frac{v}{h}\left(\frac{vt^3}{4}+\frac{2\Delta_x t^2}{3}\right)-\frac{gt^3}{8}+\frac{\Delta_x^2 t}{2h}+\frac{ht}{2} & 0 \\
0 & \frac{gt^3}{8}-\frac{ht}{2} & 0 & \frac{vt^2}{3}+\frac{\Delta_x t}{2} \\
\frac{v^2}{h^2}\left(\frac{t^4}{5}\right)+\frac{v}{h}\left(\frac{\Delta_x t^3}{2h}\right)+\frac{\Delta_x^2 t^2}{3h^2} & 0 & \frac{v^2}{h^2}\left(\frac{vt^4}{5}+\frac{3\Delta_x t^3}{4}\right)+\frac{v}{h}\left(\frac{gt^4}{10}-\frac{\Delta_x^2 t^2}{h}\right) & 0 \\
 & & +\frac{\Delta_x gt^3}{8h}-\frac{vt^2}{3}-\frac{\Delta_x^3 t}{2h^2}-\frac{\Delta_x t}{2} & \\
0 & \frac{g^2 t^4}{20}-\frac{ght^2}{3}+h^2 & 0 & \frac{gvt^3}{8}+\frac{g\Delta_x t^2}{6}-\frac{vht}{2}-\Delta_x h \\
\frac{v^2}{h^2}\left(\frac{vt^4}{5}+\frac{3\Delta_x t^3}{4}\right)+\frac{v}{h}\left(\frac{gt^4}{10}-\frac{\Delta_x^2 t^2}{h}\right) & 0 & \frac{v^2}{h^2}\left(\frac{t^4 v^2}{5}+\Delta_x vt^3+\Delta_x^2 2t^2\right)-\frac{v}{h}\left(\frac{gvt^4}{5}+\frac{\Delta_x t^3}{2h}-\frac{2\Delta_x^3 t}{h}\right) & 0 \\
+\frac{\Delta_x gt^3}{8h}-\frac{vt^2}{3}-\frac{\Delta_x^3 t}{2h^2}-\frac{\Delta_x t}{2} & & +\frac{g^2 t^4}{20}-\frac{t^2 g\Delta_x^2}{3h}-\frac{t^2 gh}{3}+\frac{2t^2 v^2}{3}+2t\Delta_x v+\frac{\Delta_x^4}{h^2}+2\Delta_x^2+h^2 & \\
0 & \frac{gvt^3}{8}+\frac{g\Delta_x t^2}{6}-\frac{vht}{2}-\Delta_x h & 0 & \frac{v^2 t^2}{3}+\Delta_x vt+\Delta_x^2
\end{bmatrix} \tag{3.71}$$

where $\Delta(t_o)=\begin{bmatrix}\Delta_x & \Delta_y & \Delta_z\end{bmatrix}^T$, $\Delta_x(t)=\Delta_x+v\cdot t$, $\Delta_y(t)=0$, $\Delta_z\equiv h$.

Additional variables are the camera focal length (f) and gravitational constant (g).

Here, the components of the specific force resolved in the body frame (f_x, f_y and f_z) are defined: $f_x=f_y=0$, and $f_z=-g$, and thus not readily apparent. Additionally, the velocity component is strictly along the *b-frame* x-axis as aircraft generally fly with no sideslip. Drift due to wind is ignored. This can be done without loss of generality as an arbitrary body frame can always be redefined to align with any constant velocity. To increase the readability of the above matrix, the tracked landmark was chosen to be along the flight path ($\Delta_y(t)=0$). Through analysis it was found that this restriction did not affect the rank of the observability grammian matrix. Following matrices are less complex and the $\Delta_y(t)=0$ restriction is omitted.

Landmarks must be tracked over time. Note that, as time goes on, the elements of the grammian grow polynomially. Additionally, the entire matrix is scaled by t (factored out, and multiplying the whole matrix). This implies the great benefit of long observation periods.

There is a suggestion of scalability as well. It is evident by the $1/h^2$ term factored out, and multiplying the whole matrix. This suggests that the absolute altitude plays as an important role as the angular geometry, and will be evaluated with simulation in Chapter 4.

To calculate the rank of the observability grammian, its Row Reduced Echelon Form (RREF) is obtained:

$$\begin{bmatrix} 1 & 0 & 0 & \frac{-\Delta_x}{v} & 0 & 0 & 0 & \frac{-g\Delta_x^{\,2}}{2v^2} - h & -\Delta_y \\ 0 & 1 & 0 & \frac{-\Delta_y}{v} & 0 & 0 & 0 & \frac{-g\Delta_x\Delta_y}{2v^2} & \Delta_x \\ 0 & 0 & 1 & \frac{-h}{v} & 0 & 0 & 0 & \frac{-gh\Delta_x}{2v^2} - \Delta_x & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & \frac{-g\Delta_y}{2v} & v \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & \frac{gh}{2v} - v & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{3.72}$$

where the position of the tracked landmark is not confined to the flight path ($\Delta_y(t) = \Delta_y$). Notice that in the RREF, all reference to time drops out. Tracking a landmark for long period does not affect rank, but will affect matrix condition number.

The 9X9 grammian matrix is not of full rank 9, but has a rank of 6 for all time. Only a partial state estimate can be obtained. Tracking a single known landmark will not enable one to aid all 9 INS states. Additional independent measurements of the navigation state are needed for full INS aiding action (full rank).

In the RREF of the grammian, the forms h/v, Δ_x/v and Δ_y/v appear quite often. The ratio of velocity to lateral displacement and ratio of velocity to altitude (*V over h,* as it is more commonly known) plays an important role in INS aiding using

bearings-only measurements. *V over h* is a measure of image apparent angular rate and has units of radians per sec (sec^{-1}). This parameter ultimately drives the measure of geometric dilution of precision (GDOP) and observability grammian condition number. By controlling *V over h*, one cannot affect rank per se, but can maximize aiding benefit and distribution.

In the case in which the object is directly over-flown ($\Delta_y = 0$), the rank remains 6 as well. This implies that the lateral position of the landmark does not affect the overall observability of the system. Geometry does, however, affect how the aiding action is distributed. Better insight to this distribution is attained by examining the condition number of the observability grammian and GDOP.

In the case in which the aircraft's trajectory is not constrained to the level, un-accelerated flight regime (where the specific force vector is aligned with the z-axis), the grammian rank increases to 7, and the RREF is

$$\begin{bmatrix} 1 & 0 & 0 & \frac{-\Delta_x}{v} & 0 & 0 & 0 & 0 & \frac{hf_y - \Delta_y f_z}{f_z} \\ 0 & 1 & 0 & \frac{-\Delta_y}{v} & 0 & 0 & 0 & 0 & \frac{\Delta_x f_z - hf_x}{f_z} \\ 0 & 0 & 1 & \frac{-h}{v} & 0 & 0 & 0 & 0 & \frac{\Delta_y f_x - \Delta_x f_y}{f_z} \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & v \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & -v\frac{f_y}{f_z} \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & -\frac{f_x}{f_z} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -\frac{f_y}{f_z} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (3.73)$$

This implies that accelerated flight does benefit INS aiding with a single tracked landmark of known position. This condition requires continuous maneuvering and is, therefore, impractical for typical missions. Accelerated flight is more favorable, but the level un-accelerated flight condition is the worst case, and will be considered from hence forth.

Tracking one landmark does not provide full observability. The question is thus posed whether tracking more objects will improve observability, and if so, how many is enough. It turns out, that two tracked landmarks are enough to provide full grammian rank, and thus, full observability and INS aiding action. The following development illustrates this.

The grammian for level, un-accelerated flight and two tracked landmarks can be expressed as the sum of two grammians, each corresponding to a single tracked landmark.

$$\mathbf{W}_2(t) = \mathbf{W}(t)\big|_{\Delta_1} + \mathbf{W}(t)\big|_{\Delta_2} \tag{3.74}$$

where, it can be expressed

$$\Delta_2 = \Delta_1 + \mathbf{P}_{\mathbf{P}1} - \mathbf{P}_{\mathbf{P}2} \tag{3.75}$$

The grammians $\mathbf{W}(t)\big|_{\Delta_1}$ and $\mathbf{W}(t)\big|_{\Delta_2}$ correspond to the two tracked landmarks, and take the form of the single track grammian above, but evaluated at the corresponding relative landmark position vectors Δ_1 and Δ_2. The observability grammian for multiple tracked landmarks simply becomes the sum of the single track grammians, evaluated for the LOS vector to the additional landmarks:

$$\mathbf{W}_\mathbf{N}(\mathbf{t}) = \sum_{i=1}^{\mathbf{N}} \mathbf{W}(\mathbf{t})\big|_{\Delta_i} \tag{3.76}$$

where N is the number of landmarks being tracked, Δ_i is the LOS vector to the i^{th} landmark, $\mathbf{W}(t)$ is the single landmark grammian from Equation (3.71), and $\mathbf{W}_\mathbf{N}(t)$ is the composite observability grammian matrix for N tracked landmarks of known location.

This realization greatly simplifies the procedure for evaluating observability grammian matrix condition number when tracking many landmarks.

For flight in the vertical plane, and without the level flight assumption, the RREF of the grammian associated with the i^{th} tracked landmark is

$$\begin{bmatrix}
1 & 0 & 0 & \frac{-\left(\Delta_x + x_1 - x_i\right)}{v} & 0 & 0 & 0 & 0 & \frac{\left(\Delta_z + z_1 - z_i\right) f_y - \left(\Delta_y + y_1 - y_i\right) f_z}{f_z} \\
0 & 1 & 0 & \frac{-\left(\Delta_y + y_1 - y_i\right)}{v} & 0 & 0 & 0 & 0 & \frac{\left(\Delta_x + x_1 - x_i\right) f_z - \left(\Delta_z + z_1 - z_i\right) f_x}{f_z} \\
0 & 0 & 1 & \frac{-\left(\Delta_z + z_1 - z_i\right)}{v} & 0 & 0 & 0 & 0 & \frac{\left(\Delta_y + y_1 - y_i\right) f_x - \left(\Delta_x + x_1 - x_i\right) f_y}{f_z} \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & v \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & -v\frac{f_y}{f_z} \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & -\frac{f_x}{f_z} \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -\frac{f_y}{f_z} \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix} \quad (3.77)$$

When two or more landmarks are tracked, the RREF of the observability grammian is an identity matrix. The question must be answered as to whether or not two tracked landmarks will always give an observability grammian with full rank. The answer is affirmative, but there are two exceptions. The grammian has full rank as long

as the LOS vectors for both landmarks are not co-linear, and the aircraft is not in a state of free-fall ($f_x = f_y = f_z = 0$).

The LOS vectors for both landmarks are co-linear if the following is true

$$\Delta_2 = a\Delta_1 \tag{3.78}$$

where *a* is a scalar multiplication factor. If the camera and the two landmarks, all lie on a straight line, the observability grammian's rank drops to 6. This is intuitive, as both landmarks provide the same measurement information to a two-dimensional camera; the only difference between the landmarks (in the camera frame) is in depth, a dimension the camera cannot discern.

The camera and two landmarks are on the same straight line in two degenerate cases: 1) the aircraft crosses the axis defined by the two landmarks, and, 2) the aircraft is flying along this same axis. In the first case, the observability drop is transitory. As the aircraft crosses this axis, the grammian rank drops momentarily, but then returns. The condition of the observability grammian matrix will remain poor until the aircraft is *far enough* away to ensure good geometry. If the aircraft travels along the axis, it will lose aiding benefit in the unobservable dimensions; position and velocity along the axis, and in rotation about it. This is illustrated in Figure 3.5.

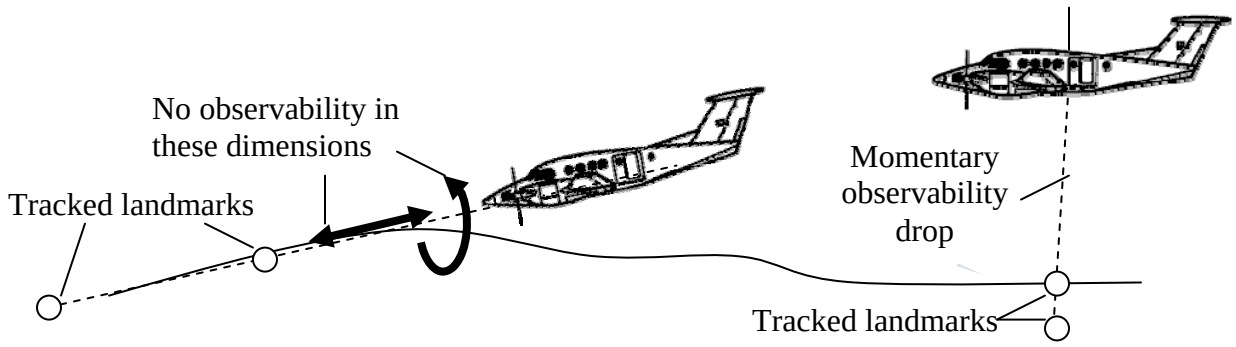

Figure 3.5: Unobservable dimensions for two landmark tracks.

An INS on board an aircraft traveling along the axis formed by the two tracked objects will receive no aiding benefit in position, velocity and rotation along and about that axis. As the aircraft approaches and crosses such an axis, observability drops and aiding suffers.

The second exception to the rule, weightlessness ($f_{x,y,z} = 0$), reduces the observability grammian matrix rank to 8 (when 2 or more landmarks are being tracked). For typical aircraft applications, weightlessness, or zero-g, conditions tend to be transitory, and should not cause significant aiding degradation. Free fall weaponry using this method of aiding would be affected, however. Notably, GPS aiding (of an INS) suffers much greater than landmark tracking; loosely coupled GPS aiding rank drops to 6 at zero-g conditions.

3.3.1.1 Non-Dimensional Study of Aiding Strength. Full observability speaks to the existence of a complete aiding solution, but says very little of the quality of that

solution. For this, GDOP and observability grammian matrix condition number must be studied. The GDOP is defined as

$$\mathbf{GDOP} = \sqrt{\mathbf{trace}\left\{\mathbf{W}(t)^{-1}\right\}} \tag{3.79}$$

where $\mathbf{W}(t)$ is the observability grammian when two landmarks are being tracked. A high condition number of the grammian matrix indicates poor geometry and consequently a high GDOP. The result of a full rank grammian with high condition number is that all states are being aided, but the aiding action is poor and unevenly distributed. GDOP provides a scalar measure for the dilution of precision attained by a particular geometry presented to the measurement device. A low GDOP value indicates little dilution of navigation aiding precision.

It is important to non-dimensionalize the variables and problem parameters when studying the nature of the observability grammian matrix. This ensures scaling issues don't obscure the insight into observability gained from the condition number or GDOP calculation, and, hence, INS aiding action. The following development explores the effects of geometry in a simple scenario.

The scenario considered entails an aircraft at an altitude *h* and velocity *v*, traveling towards, and over-flying, two landmarks arranged in a line. The positions of the landmarks are expressed as a multiple of the aircraft's altitude, as shown in Figure 3.6.

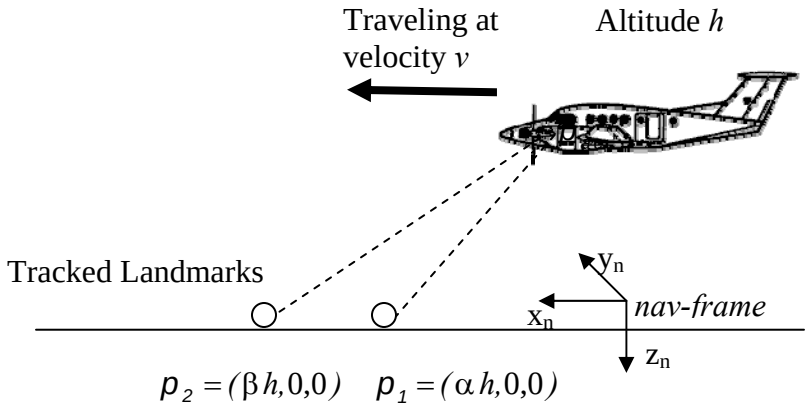

Figure 3.6: Flight in the vertical plane.

The aircraft at an altitude *h* and velocity *v* travels towards two landmarks arranged in a line; their positions are expressed as multiples of the aircraft's altitude. In this scenario, all motion is confined to the *z-x plane*; $y \equiv 0$, though 3-dimensions are still maintained in the state vector.

To non-dimensionalize, all positional variables are scaled by *h*. Time is non-dimensionalized as follows:

$$t' \to \sqrt{\frac{g}{h}}\, t \tag{3.80}$$

and

$$T' \to \sqrt{\frac{g}{h}}\, T \tag{3.81}$$

where T is the duration of flight and *g* is the Earth's gravitational acceleration.

Aircraft velocity is non-dimensionalized by setting

$$v' \to \frac{1}{\sqrt{gh}} v \tag{3.82}$$

Aircraft and ground object positions are thus expressed:

$$p_{ac} = \begin{bmatrix} v\,\mathrm{t} & 0 & 1 \end{bmatrix}^{\mathsf{T}} \qquad p_1 = \begin{bmatrix} \alpha & 0 & 0 \end{bmatrix}^{\mathsf{T}} \qquad p_2 = \begin{bmatrix} \beta & 0 & 0 \end{bmatrix}^{\mathsf{T}} \tag{3.83}$$

Finally, the specific force vector is also normalized by dividing through by g:

$$f = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^{\mathsf{T}} \tag{3.84}$$

Without loss of generality, we constrain β to be greater than α.

$$\begin{gathered} -\infty < \alpha < \infty \\ \alpha < \beta < \infty \end{gathered} \tag{3.85}$$

The parameters α and β determine the geometry of the measurement arrangement

The time variable is constrained to those instances when both ground objects are in view. For the sake of generality, assume an ideal camera with a 180 degree field-of-view (FOV) and infinite pixel resolution. The camera's tracking ability is not limited by the geometry of the objects.

Thus the four non-dimensional problem parameters are α, β, v' and T′. The 9x9 non-dimensionalized dynamics matrix for this scenario is

$$F = \begin{bmatrix} \mathbf{0}_3 & \mathrm{I}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & e_3\times \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \end{bmatrix}, \qquad e_3\times = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{3.86}$$

and the relative ground object position vectors needed for the measurement equation are

$$\Delta_1 = \begin{bmatrix} v't\text{-}\alpha & 0 & 1 \end{bmatrix}^{\mathsf{T}} \tag{3.87}$$

and

$$\Delta_2 = \begin{bmatrix} v't\text{-}\beta & 0 & 1 \end{bmatrix}^{\mathsf{T}} \tag{3.88}$$

The focal length (*f*) must also be divided by *h* so that the non-dimensional focal length is small. The nominal body-to-nav frame DCM is the identity matrix, and the altitude (*h*) is unity. Hence, the observation matrix is

$$\mathbf{H}\left(\mathbf{x}_\mathbf{c}(t)\right) = f \begin{bmatrix} 1 & 0 & \alpha - v't & 0 & 0 & 0 & 0 & \left(v't-\alpha\right)^2+1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & -1 & 0 & v't-\alpha \\ 1 & 0 & \beta - v't & 0 & 0 & 0 & 0 & \left(v't-\beta\right)^2+1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & -1 & 0 & v't-\beta \end{bmatrix} \tag{3.89}$$

Choosing the geometry

$$\alpha = 0, \qquad \beta = 1, \qquad v'=1 \quad \left(v' = \sqrt{h'g}\left[\frac{m}{sec}\right]\right), \qquad f = 10^{-5} \tag{3.90}$$

allows the GDOP and condition number ($\kappa(T)$) for this scenario to be calculated as a function of tracking time (T).

Figures 3.7 and 3.8 illustrate the behavior of non-dimensional GDOP in this simple scenario as time progresses. The amount of time a landmark is tracked is defined to be dwell time.

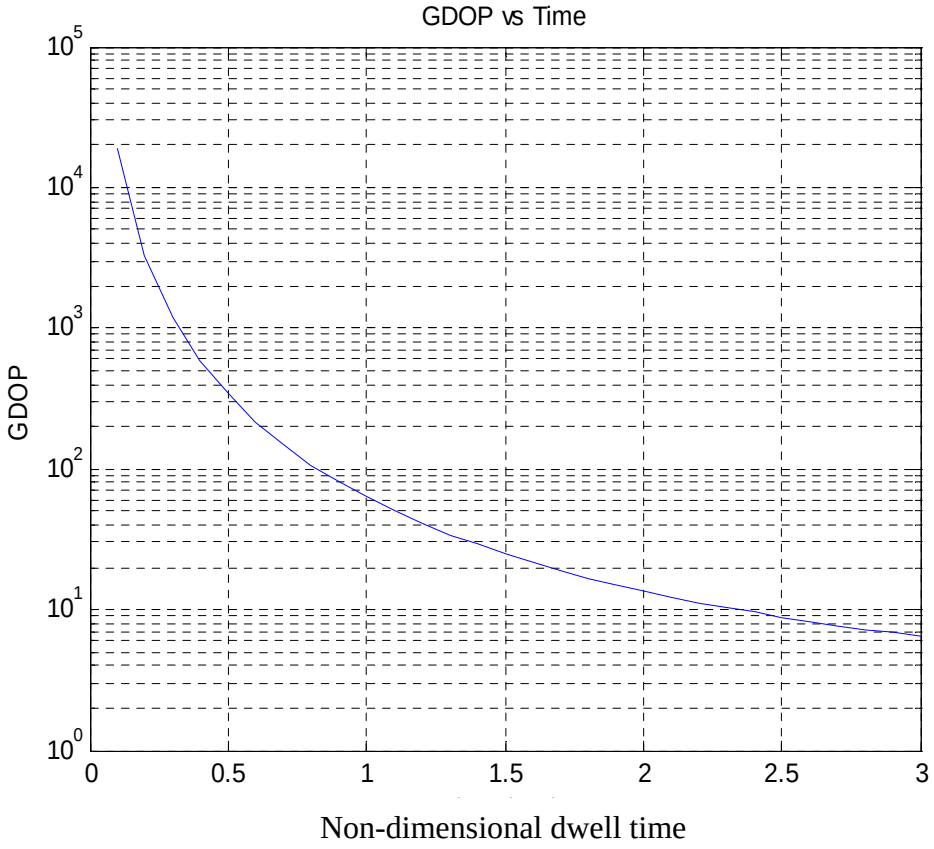

Figure 3.7: GDOP vs dwell time in the non-dimensional case study.

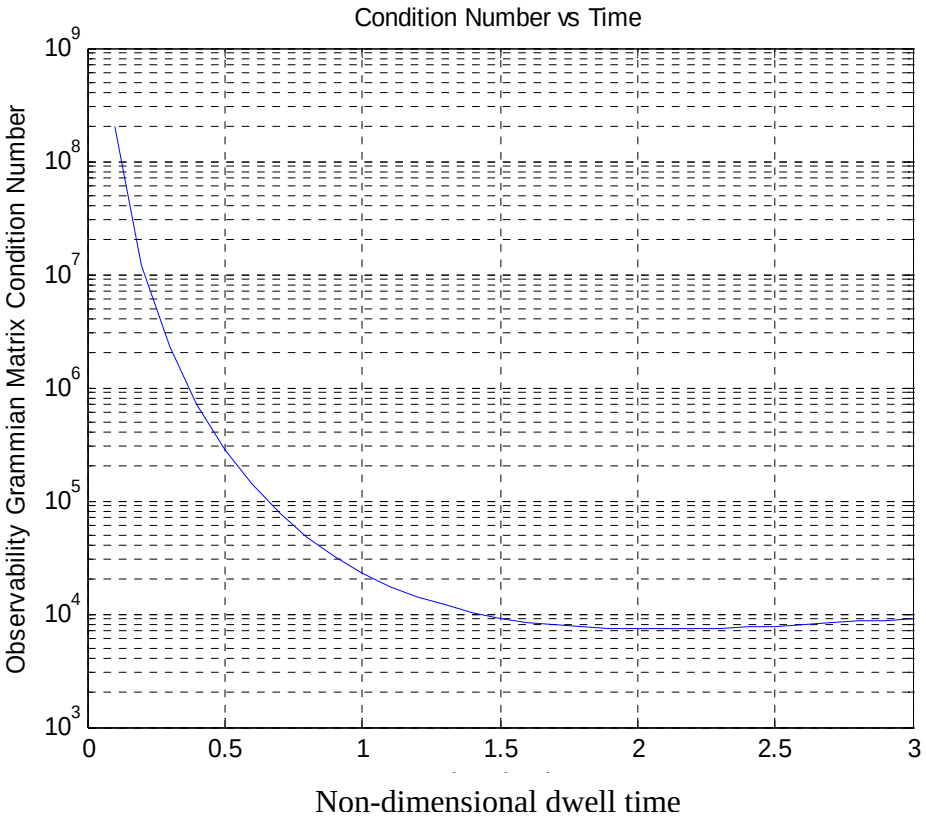

Figure 3.8: Observability grammian matrix condition number vs dwell time in the non-dimensional case study.

Figures 3.7 and 3.8 illustrate the benefit of tracking features for long periods of time. A long dwell time allows the aircraft to traverse through space, and develop larger observation or tracking angles on tracked landmarks. The nature of these plots show the great benefit of the first 45 degrees of tracking angle (occurring at time = 1 for this non-dimensionalized case), and that relatively little added benefit is gained past this point. The condition number begins to rise again after approximately 2.3, but this is attributed to

the tracked landmarks approaching the co-linear condition as they move to the horizon. In this case, the geometry effects overcome the benefit of dwell, and the matrix approaches a singular condition.

In their work, George and Sukkarieh achieved reasonable navigation performance using a SLAM algorithm [5]. They leveraged dwell time and a fast camera rate to converge on a solution. The test did, however, revisit the same environment, spiraling around a set of features with a side looking camera. Looking into the turn naturally increased dwell time. If this type of flight profile meets mission needs, then performance can be greatly increased. However, the dwell time on landmarks tracked when traveling long distances is driven by camera look angle and optical resolution. Performance is then affected by GDOP as landmarks move to the horizon. A balance must be made to optimize all variables. In either case, it is evident that tracking a landmark over time is beneficial, but there are other factors to consider.

3.3.1.2 V over h Influence on Aiding Strength. *V over h* stands out as the key aircraft controlled parameter (in the observability grammian matrix) affecting aiding quality. Landmark relative lateral displacement will change as a matter of course as landmarks traverse the camera field-of-view. The velocity-to-height ratio, however, is directly established by the flight path. The observability grammian matrix is a nine-dimensional function making a concise analysis very difficult. One dimensional measures of GDOP and matrix condition number give some insight, but sweeping generalizations cannot be made. The *V over h* investigation is even more complex as varying velocity, height or dwell time, ultimately affects the geometry. No true single

variable isolation can be made. In any case, this section will examine the non-dimensional scenario in an attempt to gain some insight.

Consider the non-dimensional case describe above, but allow velocity to vary, adjusting the *V over h* ratio. Because dwell time and tracking angle are related by velocity, two conditions are considered. In the first, the dwell time is held constant as velocity (*V over h*) is varied. This leads to a tracking angle that increases with velocity (*V over h*). In the second, the total tracking angle is made constant by varying the dwell time to achieve the same distance traveled. In this case, one landmark starts 45 degrees in front and finishes directly below, while the other starts directly below and finishes 45 degrees behind. The constant dwell time case is important because time between images (measurements) is constant. Since the Kalman filter aims to limit INS error growth that accumulates between measurements, it is important to examine the ability of the system to control this growth effectively over constant time intervals. These intervals will of course vary with application. The constant tracking angle case is important to examine as it eliminates the geometry from the equation, leaving the effect of dynamics.

The effect of scalability cannot be ignored. It is also illustrated for each of the conditions describe above. The height is varied about the nominal height (ho) for three different cases, but angular geometry is held constant. Figure 3.9 illustrates the effect varying *V over h* has on observability grammian matrix condition number.

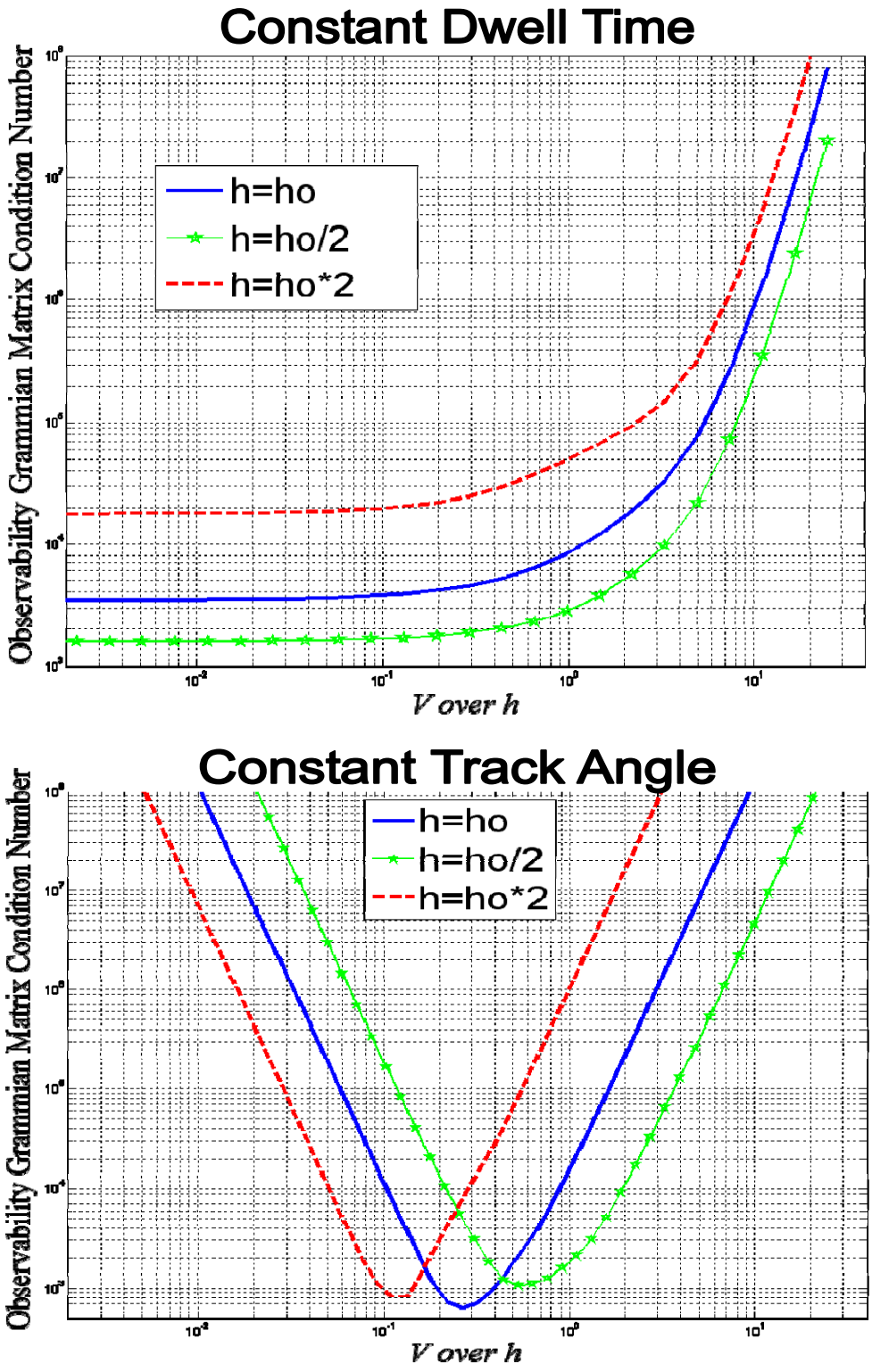

Figure 3.9: Observability grammian matrix condition number as a function of *V over h*. For a non-dimensionalized case, as altitude (h) varies about a nominal height (ho).

For the constant dwell time case, lower *V over h* ratios yield lower condition numbers, and thus more evenly distributed aiding (among all estimated states). A ratio of approximately 1 appears to be a knee in the curve before the matrix condition number grows very large. This is due to the fact that, for the non-dimensionalized dwell time of 1, both landmarks remain within 45 degrees of the nadir when the *V over h* ratio is less than 1. The dwell time study above showed that tracking angles up to 45 degrees provide the best aiding. For *V over h* ratios greater than 1, the landmarks have passed underneath and proceeded towards the horizon. In this case, the geometry has drastically changed and the landmarks approach a co-linear condition. This is the reason condition number grows, not *V over h* ratio itself. When the landmarks remain inside 45 degrees of nadir, *V over h* has little to no effect on grammian matrix condition number or the distribution of aiding among the navigation state variables.

Notably, a lower altitude (h=ho/2) provides better aiding than a higher one (h=ho*2), despite the same *V over h* ratio and geometry (absolute altitude drives the magnitude of the condition number, while *V over h* drives the shape of the graph). This speaks to the scalability problem faced by bearings-only measurements. The angular resolution of a camera drives its ability to discern small displacements from far away, and thus positional estimation suffers. This study implies that flying slow and low (flying lower for a given *V over h* ratio) provides the strongest and most even aiding. This is supported by Veth's work on indoor navigation [26].

In the constant track angle case, the landmarks all remain inside 45 degrees of nadir, and each experience a 45 degree tracking angle. The dwell time however varies

with *V over h*. Figure 3.9 implies a *sweet spot* somewhere in the middle. The *V over h* ratio factors into the observability grammian in many places. Too large or small a value will naturally increase the condition number of the matrix. This finding merely indicates that the aiding will not be evenly distributed. It is proposed that landmarks which have low angular rates (or effective *V over h*) provide good attitude aiding, but poor positional aiding. It is also proposed that landmarks which have the highest angular rates (or effective *V over h*) provide the best positional aiding and poorest attitude aiding. This proposition is supported by the fact that a star tracker (stars are effectively all on the horizon because none exhibit angular rates due to aircraft translation) provides excellent attitude aiding, but no positional aiding. The fact that Figure 3.9 illustrates a non-dimensional case and the *sweet spot* varies with absolute height means that predicting the *sweet spot* is very difficult. An aiding strategy that maximizes *V over h* span will increase the likelihood that the *sweet spot* is captured. Also, if in fact the two ends of the spectrum represent good attitude aiding and good positional aiding respectively, then capturing both ends will complement each other. Chapter 4 will experimentally evaluate the effect of *V over h* and attempt to validate these propositions.

A low *V over h* ratio at low altitude appears to be the most favorable condition to provide evenly distributed aiding to the whole navigation state. The added benefit of increased attitude aiding is felt throughout the system, as a good attitude estimate is critical to resolve the rest of the navigation state. This proposal is supported by the work of George and Sukkarieh who flew a small UAV at approximately 125 meters altitude and at slow speeds, looking sideways (into the turn) while orbiting a target area. Altitude

and *V over h* were inherently low, and they achieved reasonable performance [5]. However, this condition may not be attainable for many applications which travel in a straight line.

The full aiding picture cannot be seen until GDOP is examined. Consider the non-dimensional scenario from a GDOP perspective; see Figure 3.10. Again, the constant dwell time and tracking angle are examined as well as scalability.

In the constant dwell time case, there appears to be a slight advantage to a *V over h* ratio between 1 and 10, despite the absolute altitude. In this region, the landmarks have exited the 45 degree cone about nadir. The condition number has increased, but the GDOP has decreased. This finding is not intuitive, but suggests that perhaps the higher tracking angle (for the given dwell time) provides an overall benefit that overcomes the effects that penalize condition number. Thus, there is yet another sweet spot, but this one at a higher *V over h* than the associated sweet spot described above. This suggests that an aiding strategy that captures higher *V over h* ratios could take advantage of an improved GDOP.

Once again, the scalability principle shows that absolute altitude significantly affects GDOP across the board. Lower absolute altitudes reduce precision dilution, and thus improve aiding action.

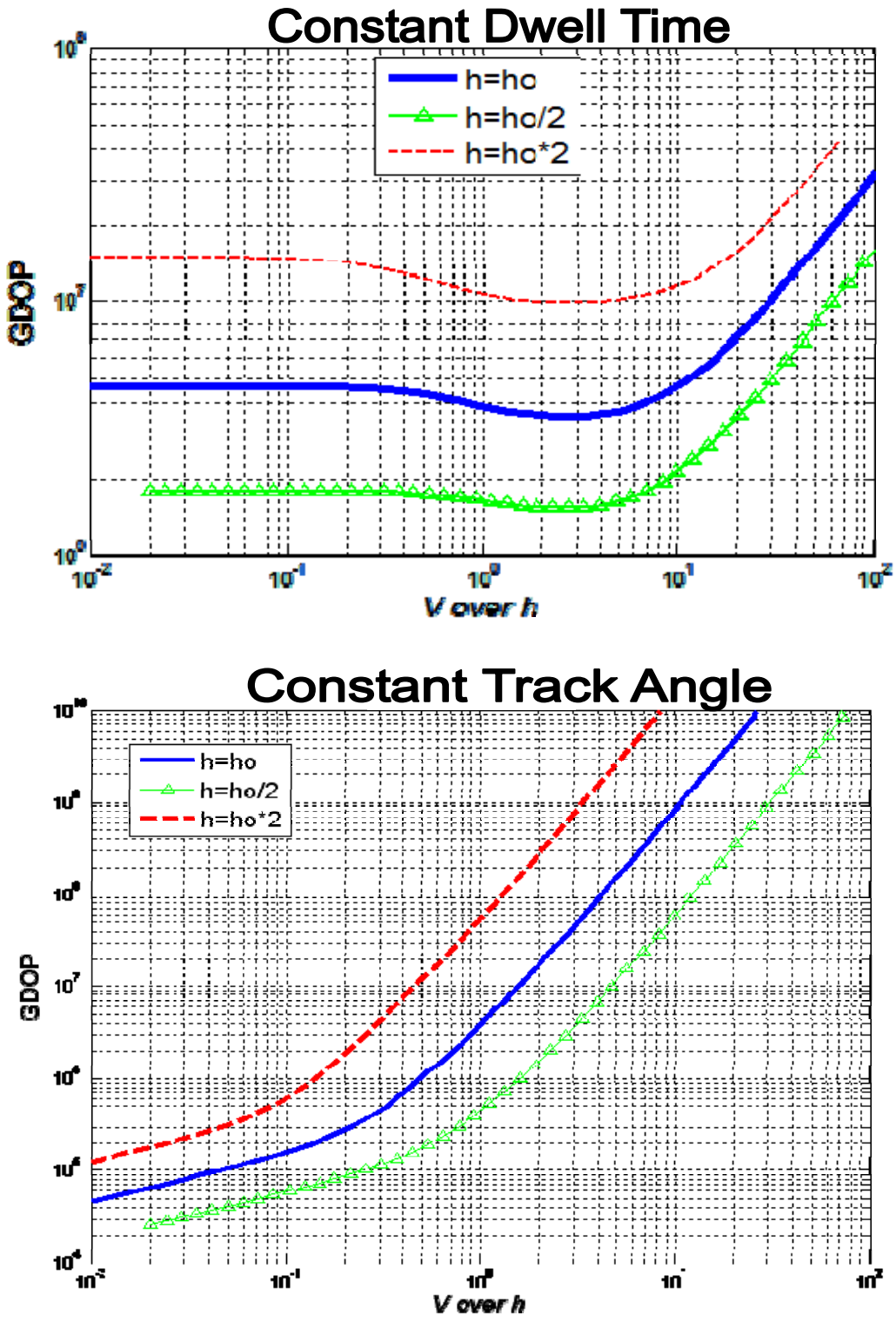

Figure 3.10: Geometric Dilution of Precision as a function of V over h. For a non-dimensionalized case, as altitude (h) varies about a nominal height (ho).

In the constant track angle case, a lower *V over h* definitively improves GDOP. Since the aircraft-to-landmark geometry dependence is removed, it isn't readily apparent why this is the case. If the discrete case is considered, a lower *V over h* allows more measurements in the same tracking angle. More measurements allow the corrupting errors to be averaged out more precisely. If the limit is taken such that a continuous set of measurements is available, then the benefit of a lower *V over h* is understandable. The dynamics of the inertial system cannot be ignored either. In any case, this finding suggests that an aiding strategy that captures lower *V over h* ratios could take advantage of an improved GDOP.

So far there has been evidence that capturing both high and low *V over h* ratios provide benefit. This cannot be accomplished with a single landmark, but can be done by tracking many different landmarks simultaneously. *V over h* ultimately drives aiding strength and distribution, and will be evaluated experimentally in Chapter 4.

3.3.1.3 Look Angle Influence on Aiding Strength. The question then arises, what *V over h* and what altitude should be flown for a given application (aircraft and mission). This decision, of course, will be influenced by the performance of the aircraft and its mission requirements, but *V over h* can be affected another way. Given an aircraft flying over flat terrain, the *V over h* ratio described only applies at nadir. In actuality, *V over h* should be considered a measure of apparent angular rate, which varies throughout the scene. Objects on the horizon show zero angular rate (for straight and level flight), while *V over h* is highest at nadir. The benefit of a lower *V over h* is shown in Figures 3.9 and 3.10. The portion of the scene near the horizon naturally exhibits low *V over h*.

The portion of the scene near nadir naturally exhibits higher *V over h*, but doesn't necessarily approach the critical and unknown *sweet spots*. *V over h* can be, to an extent, varied by observing different portions of a scene.

Suppose that the entire scene exhibits a *V over h* less than 1 sec^{-1}. This is optimal for even distribution of aiding (for constant time intervals), but can't be generally guaranteed. However, there exists a portion of every scene, between nadir and the horizon, where *V over h* is less than this threshold. It must be stressed that condition number suffers if the lowest *V over h* is caused by tracking landmarks on the horizon. These landmarks will never generate large tracking angles. With a wide enough lens, the camera FOV should strive to capture portions near the horizon for even aiding strength to all navigation states (over constant time intervals) in order to ensure that all navigation state errors are constrained. However, due to the naturally low tracking angles of these landmarks, a complementary approach that tracks landmarks over large angles must be accomplished.

If a landmark position requires triangulation, a lower GDOP is more favorable. This was shown to happen between a *V over h* of 1 and 10 sec^{-1} (for the non-dimensional case). This condition cannot be guaranteed in any scene, but there is no penalty for trying. Given conventional aircraft performance, this condition would rarely exist anywhere other than at nadir. Also, it would be rare for a conventional aircraft to exceed a *V over h* of 3-5 sec^{-1} at nadir. Thus, to ensure the maximum benefit of a low GDOP, a camera FOV should strive to capture nadir. Since a low *V over h* can always be guaranteed, and a high *V over h* cannot, capturing nadir should outweigh capturing the

horizon if camera field-of-view is limited, especially since landmarks tracked on the horizon require another to be orthogonally tracked at high *V over h*.

The purposed *sweet spots* in the *V over h* spectrum cannot be practically predicted or controlled. Despite this nature, it has already been shown that capturing both high and low *V over h* potions of the scene is beneficial. Fortunately the *sweet spots* reside between the high and low limits demonstrated in the non-dimensional case. An aiding strategy that captures the horizon and nadir will maximize the potential to capture the *V over h sweet spots* described here.

Since landmarks generally exist below the horizon, a 180 degree field-of-view camera lens could reasonably view the entire scene presented to an aircraft. This, however, creates a complex calibration problem. A 90 degree field of view is a practical compromise between pin-hole camera model calibration, and capturing enough of the scene. This would also ensure capturing nadir and one horizon, but which horizon is optimal? Humans look forward, capturing the forward horizon, but this is because humans use their eyes for navigation and guidance. From an aviation navigation standpoint, guidance can be accomplished by steering to inertial coordinates, freeing the camera to look anywhere (if dedicated to the navigation task).

An EKF is an iterative process, and the element of time ordering plays an important role. The method by which a landmark enters and exits the scene affects the resultant aiding. If a landmark's position is well surveyed, then, arguably, order doesn't matter. One would want simply to track it for as long as possible, or *dwell* on it, with the lowest possible observability grammian condition number and GDOP to ensure the most

optimal aiding. Suppose, however, that the landmark position is not known. It would have to be located by triangulation before dwelling on it would provide any benefit. This suggests that the camera should be positioned to observe landmarks in the higher *V over h* portion of the scene before dwelling in the lower *V over h* portion of the scene. For a 90 degree field-of-view, this means pointing the camera backwards at a 45 degree angle. Figure 3.11 illustrates this principle.

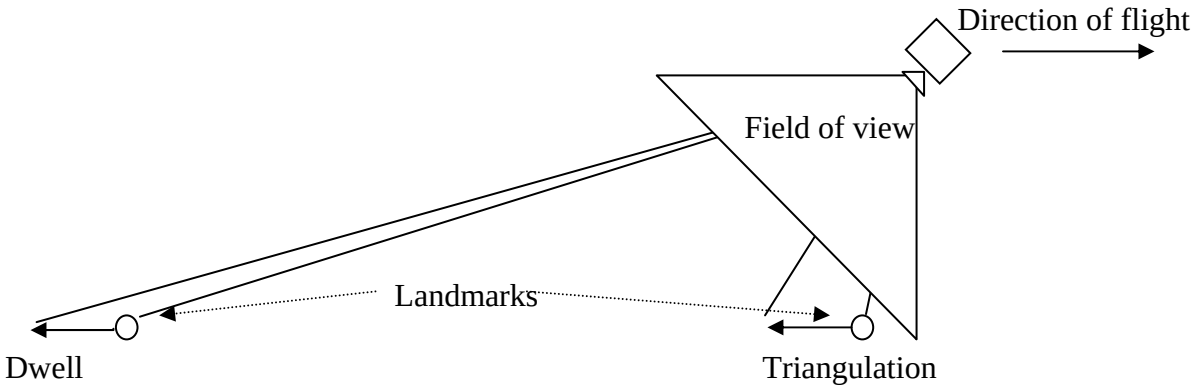

Figure 3.11: Rear facing camera geometric advantage.

A rearward looking camera allows landmarks to be triangulated before they are tracked, thus maximizing *useful dwell time*. This assertion is counter-intuitive, but insightful and unique to this research.

If time were not a factor, i.e., an offline optimal smoother or batch estimator was used, order of flow would not matter. However, it was shown in Figure 3.7 that dwell time affects condition number. Looking backwards maximizes *useful dwell time* on features that have been first triangulated. Looking forward, the system dwells on

landmarks before they have been triangulated well (the best GDOP occurs just before landmarks leave the scene). This is of course a simplification of a continuous process, but the principles remain. An experimental evaluation of camera look angle is accomplished in Chapter 4, supporting this assertion.

3.3.1.4 Scene Geometry Influence on Aiding Strength. From a strict observability standpoint, one needs only to track two surveyed landmarks. Matrix condition and GDOP, however, depend on geometry. Tracking many *randomly placed* landmarks increase the probability of attaining good geometry (a low condition number and good GDOP) at all times. The observability grammian, condition number, and GDOP calculations are highly non-linear, complex, time-variant, and thus difficult to predict. A single study of the effects of scene geometry provides little predictive value for deciding which geometry would best suit the myriad of scenarios an aircraft could face. In practice, one would benefit from an over-determined system, by tracking as many landmarks as possible, providing a varied set of geometry at all times. Using statistical brute force, randomly selected landmarks spanning the scene give the most robust strategy to ensure a low GDOP and condition number at all times.

3.3.2 Alternative Measurements to Augment Observability. Reconsider the single landmark tracking case. It is an under-determined problem, resulting in a less-than-full-rank observability grammian matrix. The question is posed, what, if any, other passive measurements may be made to achieve full rank? This investigation is pivotal in the development of an aiding strategy for tracking landmarks of opportunity.

An inspection of the observability grammian's RREF in Equation (3.72) reveals that measuring h eliminates ambiguities between height and velocity (allowing the h/v dimension to be resolved). Both position and velocity would benefit since this term appears in row 3 and 6, suggesting a potential rank increase of 2. Additionally, the specific force vector components ($f_{x,y,z}$) are factored with $\Delta_{x,y}$ and v, requiring a measurement to resolve. In level flight $f_{x,y}$ are essentially 0, leaving only v and the horizontal LOS components ($\Delta_{x,y}$) in the last column. The most immediately evident solution is to measure heading (ψ) in order to resolve directional ambiguities. This suggests that, in lieu of tracking additional landmarks, full observability may be achievable with passive measurements that are readily available on most aircraft: altitude via an altimeter, and heading via a compass.

Taking an altitude measurement results in augmenting the observation matrix as such:

$$\mathbf{H}'\left(\mathbf{x}_c(t)\right)=\begin{bmatrix}\mathbf{H}\left(\mathbf{x}_c(t)\right)\\ \begin{bmatrix}0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0\end{bmatrix}\end{bmatrix} \tag{3.91}$$

where $\mathbf{H}\left(\mathbf{x}_c(t)\right)$ is defined in Equation (3.59). Recalculating $\mathbf{W}(t)$ reveals that the simple addition of an altitude measurement increases the rank of $\mathbf{W}(t)$ from 6 to 8. The RREF becomes:

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\Delta_y \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & \Delta_x \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & v \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{3.92}$$

Notably, when the un-accelerated flight restriction is removed, full rank is achieved. Maneuvering has consistently been shown to help aiding.

Likewise, taking a heading measurement results in augmenting the observation matrix as such:

$$\mathbf{H}'\left(\mathbf{x}_c(t)\right) = \begin{bmatrix} \mathbf{H}\left(\mathbf{x}_c(t)\right) \\ [0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 1] \end{bmatrix} \tag{3.93}$$

Recalculating $\mathbf{W}(t)$ reveals that the simple addition of a heading measurement increases the rank of $\mathbf{W}(t)$ from 6 to 7. The RREF becomes:

$$\begin{bmatrix} 1 & 0 & 0 & \frac{-\Delta_x}{v} & 0 & 0 & 0 & \frac{-g\Delta_x^{\,2}}{2v^2} - h & 0 \\ 0 & 1 & 0 & \frac{-\Delta_y}{v} & 0 & 0 & 0 & \frac{-g\Delta_x\Delta_y}{2v^2} & 0 \\ 0 & 0 & 1 & \frac{-h}{v} & 0 & 0 & 0 & \frac{-gh\Delta_x}{2v^2} - \Delta_x & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & \frac{-g\Delta_y}{2v} & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & \frac{gh}{2v} - v & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{3.94}$$

Notably, when the un-accelerated flight restriction is removed, a rank of 8 is achieved. Maneuvering has consistently been shown to help aiding.

Lastly, taking both a heading and altitude measurement results in augmenting the observation matrix as such:

$$\mathbf{H}'\left(\mathbf{x}_c(t)\right) = \begin{bmatrix} \mathbf{H}\left(\mathbf{x}_c(t)\right) \\ \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \\ \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \end{bmatrix} \tag{3.95}$$

and produces a full rank observability grammian matrix.

Notably, measuring either Δ_x or Δ_y (aircraft position states) yield the same result as measuring h, but are not measurements that can be made passively like altitude. Single channel inertial velocity measurements increase rank by 1, but cannot be measured by typical (independent) passive sensors. Additionally, they have no effect on rank when altitude is simultaneously being measured. Roll or pitch measurements have the same effect as heading, but are again not available through (independent) passive sensors. Lastly, only heading or roll measurements solve the observability problem caused by the zero-g condition, restoring full rank.

In summary, independent measurements of the heading (ψ), the altitude (h), and bearings measurements of a single known landmark make it possible to update the complete nav-state. Atmospheric pressure bias and magnetic variation would need to be considered, but change slowly and should provide little challenge. This is an important finding as the additional measurements of ψ, h will be further discussed in the case when landmark position is not known independently, and full observability cannot be achieved through landmark bearing measurements alone.

3.3.3 Estimated Landmark Position Tracking Observability. Tracking surveyed landmarks (without augmenting the navigation state vector to estimate their position) is a relatively trivial task and has been demonstrated before [8]. When landmark positions must be estimated (as proposed in SLAM and SLAAMR), the navigation state dimension grows and the observability grammian changes. As described in Section 3.2.6, when tracking N landmarks of unknown position or landmarks of opportunity (LOO), the

navigation and error states are augmented, and the observation matrix $\mathbf{H}(\mathbf{x_c}(t))$ is augmented with two additional rows for each of the N tracked LOO:

$$\mathbf{H}(\mathbf{x_c}(t)) = \begin{bmatrix} \mathbf{H}_1(\mathbf{x_c}(t)) \\ \vdots \\ \mathbf{H_N}(\mathbf{x_c}(t)) \end{bmatrix}, \qquad \Delta_i = \mathbf{T}_{en}(\mathbf{p}_{tgt\mathbf{i}} - \mathbf{p}) - \mathbf{C}_b^n \mathbf{p}_{cam}^b \quad , \qquad i = 1, \ldots, N$$

$$\mathbf{H}_i(\mathbf{x_c}(t)) = \frac{f}{\left(\Delta_i^{\mathsf{T}} \mathbf{C}_c^n \mathbf{e}_3\right)^2} \begin{bmatrix} -\Delta_i^{\mathsf{T}} \mathbf{C}_c^n \mathbf{E}_2 \mathbf{C}_n^c & \vdots & \mathbf{0}_{1\mathbf{x}3} & \vdots & \Delta_i^{\mathsf{T}} \mathbf{C}_c^n \left(\mathbf{e}_3 \Delta_i^{\mathsf{T}} \left(\mathbf{C}_c^n \mathbf{e}_1\right)\times - \mathbf{e}_1 \Delta_i^{\mathsf{T}} \left(\mathbf{C}_c^n \mathbf{e}_3\right)\times\right) \\ +\Delta_i^{\mathsf{T}} \mathbf{C}_c^n \mathbf{E}_1 \mathbf{C}_n^c & \vdots & \mathbf{0}_{1\mathbf{x}3} & \vdots & \Delta_i^{\mathsf{T}} \mathbf{C}_c^n \left(\mathbf{e}_3 \Delta_i^{\mathsf{T}} \left(\mathbf{C}_c^n \mathbf{e}_2\right)\times - \mathbf{e}_2 \Delta_i^{\mathsf{T}} \left(\mathbf{C}_c^n \mathbf{e}_3\right)\times\right) \\ & \vdots & \mathbf{0}_{1\mathbf{x}(3\bullet(i-1))} & \vdots & +\Delta_i^{\mathsf{T}} \mathbf{C}_c^n \mathbf{E}_2 \mathbf{C}_n^c \quad \vdots \quad \mathbf{0}_{1\mathbf{x}(3\bullet(N-i))} \\ & \vdots & \mathbf{0}_{1\mathbf{x}(3\bullet(i-1))} & \vdots & -\Delta_i^{\mathsf{T}} \mathbf{C}_c^n \mathbf{E}_1 \mathbf{C}_n^c \quad \vdots \quad \mathbf{0}_{1\mathbf{x}(3\bullet(N-i))} \end{bmatrix} \tag{3.96}$$

When the observability grammian matrix is calculated with the new observation matrix and state vectors, it takes a similar form as in the known tracking case. The inner 9x9 matrix is identical

$$\mathbf{W}_\mathbf{N}^{'}(t)_{9\mathbf{x}9} = \mathbf{W}_\mathbf{N}(t) \tag{3.97}$$

where $\mathbf{W}_\mathbf{N}^{'}(t)$ is the observability grammian matrix for N estimated landmark tracks and $\mathbf{W}_\mathbf{N}(t)$ is the observability grammian matrix from the N known landmark tracking case as defined in Equation (3.76).

Additionally, the summation property from Equation (3.76) remains true.

$$\mathbf{W}_{\mathbf{N}}^{'}(\mathbf{t}) = \sum_{i=1}^{\mathbf{N}} \mathbf{W}_{i}^{'}(\mathbf{t})\Big|_{\Delta i} \tag{3.98}$$

and $\mathbf{W}_{i}^{'}(\mathbf{t})\Big|_{\Delta i}$ is defined

$$\mathbf{W}_{i}^{'}(\mathbf{t})\Big|_{\Delta i} = \begin{bmatrix} \mathbf{W}_{i} & \mathbf{0}_{(3\cdot(i-1))\mathbf{x}9} & -\mathbf{W}_{i(1:9,1:3)} & \mathbf{0}_{9\mathbf{x}(3\cdot(\mathbf{N}-i))} \\ \mathbf{0}_{(3\cdot(i-1))\mathbf{x}9} & \mathbf{0}_{(3\cdot(i-1))\mathbf{x}(3\cdot(i-1))} & \mathbf{0}_{3\mathbf{x}3} & \mathbf{0}_{3\mathbf{x}3} \\ -\mathbf{W}_{i(1:3,1:9)} & \mathbf{0}_{3\mathbf{x}3} & \mathbf{W}_{i(1:3,1:3)} & \mathbf{0}_{3\mathbf{x}3} \\ \mathbf{0}_{(3\cdot(\mathbf{N}-i))\mathbf{x}9} & \mathbf{0}_{3\mathbf{x}3} & \mathbf{0}_{3\mathbf{x}3} & \mathbf{0}_{(3\cdot(\mathbf{N}-i))\mathbf{x}(3\cdot(\mathbf{N}-i))} \end{bmatrix} \tag{3.99}$$

where $\mathbf{0}_{m\mathbf{x}n}$ is a *m*-by-*n* matrix of zeros, $\mathbf{W}_{i}$ is short hand for $\mathbf{W}_{i}(\mathbf{t})\Big|_{\Delta i}$ and $\mathbf{W}_{i(j:k,m:n)}$ is a subset of $\mathbf{W}_{i}$ defined by the *j* through k^{th} row and *m* through n^{th} column. It stands to reason that a similar extension is made for the RREF of $\mathbf{W}_{\mathbf{N}}^{'}(t)$.

Two questions arise and must be addressed. First, can full observability grammian matrix rank be obtained when no survey is provided? Second, does the increased navigation state dimension affect observability when landmark positions are surveyed independently, but inserted as "good" estimates? The answer to the first question is no, and the second is yes, but for all practical purposes, no. The development follows.

3.3.3.1 Landmark of Opportunity Tracking. Again, consider the case of un-accelerated flight with two tracked landmarks, with positions that are now estimated. Call these landmarks of opportunity (LOO). $\mathbf{W}_{1}^{'}(t)$ becomes a 15x15 matrix with rank

10. Understandably, full rank cannot be achieved by measuring angular differences between two positions, both of which are unknown. In fact, for the specific case of level, un-accelerated flight over flat terrain, rank trails state dimension by a constant value, 5, as the number of tracked landmarks increases. State dimension N_x grows linearly with number of tracked landmarks N_L as follows

$$N_x = N_{x_0} + 3 \cdot N_L ,$$

$$\text{Rank}\left(\mathbf{W}_{\mathbf{N}}'(t)\right) = N_x - 5 \tag{3.100}$$

where N_{x_0} is the length of the basic navigation state (9 or 15, depending on inclusion of sensor bias states). This constant offset suggests that tracking *many* landmarks may eventually dilute the effect of the underdetermined set of measurement equations. The dimensions along which observability is not obtained are not apparent, but their number does not grow with added landmark tracks. It cannot be said that tracking more LOO will alter the nature of the unobserved dimensions, but there is no observability penalty for tracking many. It is proposed that tracking many LOO may dilute the effect of deficient rank with no penalty.

This proposition is key to the design of the SLAAMR algorithm, as it incorporates many simultaneous tracks. The effects of tracking varying numbers of LOO will be evaluated experimentally in Chapter 4. Tracking more landmarks increases state dimension linearly, but computation speed geometrically. For the SLAAMR algorithm, implemented in Matlab©, using ten tracks was a practical compromise between speed and performance. Ten tracks were sufficiently many, yet allowed real-time computation.

Another important finding is that, if both tracked landmarks fall along the flight path, rank drops from 10 to 9. This is an important distinction from the known landmark tracking case in which lateral offset ($\Delta_y(t) \neq 0$) did not affect rank. This speaks to the importance of varied landmark geometry. Along with tracking many landmarks, SLAAMR ensures to maximize lateral separation between chosen landmarks. To ensure a robust solution allowing for rolls and turns, new landmarks are chosen to track such that the entire population of tracked features spans the camera frame. This allows rolls and turns in any direction without dropping a significant portion out of the field of view. SLAAMR accomplishes this by weighting new feature selection based on distance from each other, frame edges, open space in the image, etc. This is an ad hoc technique and designed with engineering judgment.

When the additional independent measurements of altitude and heading are applied as described in Section 3.3.2 to the case in which two unknown landmarks are tracked, rank increases from 10 to 12 (full rank being 15); each measurement contributing one order of rank. In fact, rank now trails state dimension by only 3 as the number of tracked landmarks increases. This holds true when the level flight restriction is removed. In the known landmark position tracking case, heading and altitude measurements enabled full observability when tracking a single landmark. In the unknown case, full rank can never be achieved. Nonetheless, observability increases, and the effects of these measurements is evaluated experimentally in Chapter 4.

3.3.3.2 Using DTED to Constrain Errors Further. Digital Terrain Elevation Data (DTED) is a satellite-based survey of the Earth's surface. It can be used to resolve

uncertainty in a LOO's vertical position. A direct measurement cannot be taken, but intersecting the 3-dimentional DTED map with the LOO's image projection provides a good estimate of elevation. With perfect knowledge of aircraft position and attitude, the errors in a DTED-based LOO elevation estimate are dependent on the fidelity of the DTED map itself. However, if the aircraft position and attitude estimate used to intersect the map are errant, the wrong portion of the map will be intersected, and the estimate will be in error. Areas of flat terrain are naturally less prone to this source of error, while mountainous regions are more prone. In either case, without perfect knowledge of aircraft navigation state, DTED intersection errors are not independent of the navigation state errors, nor are they zero-mean or Gaussian.

Consider, however, that perfect navigation state knowledge was obtainable, and two LOO were being tracked and their positions estimated. The observability contribution of incorporating DTED intersections should be studied. The DTED intersection produces a 3-dimensional position estimate, but only the vertical element is independent of the navigation state estimates. If DTED intersections could be considered independent and direct measurements of landmark vertical position, the observation matrix would be augmented as such:

$$\mathbf{H}'\left(\mathbf{x}_c(t)\right)=\begin{bmatrix}\mathbf{H}\left(\mathbf{x}_c(t)\right)\\ \begin{bmatrix}0 & 0 & \cdots & 0 & 0 & 1 & 0 & 0 & 0\end{bmatrix}\\ \begin{bmatrix}0 & 0 & \cdots & 0 & 0 & 0 & 0 & 0 & 1\end{bmatrix}\end{bmatrix} \tag{3.101}$$

where the third (vertical) position state of both LOO are directly measured. The resultant observability grammian matrix rank increases by an order of 1, no matter how many LOO are being tracked. This holds true whether one LOO vertical position is estimated with DTED or all are.

This behavior also extends to the case in which heading and aircraft altitude are measured, causing rank to trail state dimension by only 2 as the number of tracked landmarks increases. This study is ideal, and not truly realizable, but the practical effect of DTED intersection is experimentally evaluated in Chapter 4.

3.3.3.3 Surveyed Landmarks with Augmented State Vector. Consider that the tracked landmarks were independently surveyed, but done so imperfectly. In essence, the landmark positions are estimated, but the uncertainty is small, and the source is independent of the system being aided. The independent position measurement/survey is applied directly to the landmark position states, and the uncertainty applied to the error state covariance matrix. The state vector is the collection of estimated states augmented with the landmark's surveyed position ($\mathbf{p}_{\mathbf{survey}}$):

$$\mathbf{x}(t_i) = \left[\hat{\mathbf{p}}_{ac}^{\mathsf{T}} \quad \hat{\mathbf{v}}_{eb}^{n\,\mathsf{T}} \quad \hat{\Psi}_{nb}^{n\,\mathsf{T}} \quad \vdots \quad \hat{\mathbf{p}}_{tgt1}^{\mathsf{T}} \quad \cdots \quad \mathbf{p}_{\mathbf{survey}}^{\mathsf{T}} \quad \cdots \quad \hat{\mathbf{p}}_{tgt\mathbf{N}}^{\mathsf{T}} \right]^{\mathsf{T}} \tag{3.102}$$

and the associated 3-dimensional uncertainty ($\mathbf{P}_{survey}$) is inserted into the whole state covariance matrix ($\mathbf{P}(t_i)$), with zeros filling the cross covariance elements:

$$\mathbf{P}(t_i) = \begin{bmatrix} \mathbf{P}(t_i)_{1:j,1:j} & \mathbf{0}_{j\text{x}3} & \mathbf{P}(t_i)_{1:j,(N-j+4):N} \\ \mathbf{0}_{3\text{x}j} & \mathbf{P}_{survey} & \mathbf{0}_{3\text{x}(N-j+4)} \\ \mathbf{P}(t_i)_{(N-j+4):N,1:j} & \mathbf{0}_{(N-j+4)\text{x}3} & \mathbf{P}(t_i)_{j+4:N,j+4:N} \end{bmatrix} \tag{3.103}$$

where $\mathbf{P}(t_i)_{j:k,j:k}$ is a subset of the whole covariance matrix $\mathbf{P}(t_i)$, and $\mathbf{0}_{j\text{x}k}$ is a jxk matrix of zeros.

Because the landmarks are assumed to be stationary, their positional uncertainty volume does not grow with time (effective portion of state transition matrix $\Phi(t,t_0)$ is identity). In fact, measurements on these injected landmarks only serve to better the initial surveyed position estimate. If the initial positional uncertainty of the surveyed landmark is small enough (relative to other error states), the Kalman filter will treat it as if it is *near* truth. Most of the correction derived from measuring bearings to the landmark will be distributed to the navigation states. In fact, as the uncertainty of the surveyed position approaches zero ($\mathbf{P}_{survey} \to \mathbf{0}_{3x3}$), the landmark position states become completely independent, and may be eliminated from the EKF. This proposition is supported by examining the Kalman filter gain in two non-dimensionalized cases.

Consider an aircraft traveling at unit velocity, at unit height, tracking a single landmark displaced a unit distance laterally and longitudinally. All navigation states have identical, unit valued, uncertainties ($\mathbf{P}$=Identity). The Kalman filter gain ($\mathbf{K}$) is

$$\mathbf{K} = \begin{bmatrix} 0 & .13 & 0 & 0 & 0 & 0 & -.13 & 0 & .25 & 0 & -.13 & 0 \\ .028 & 0 & -.056 & 0 & 0 & 0 & 0 & .14 & 0 & -.028 & 0 & .056 \end{bmatrix}^{\mathsf{T}}$$

Note that both aircraft position (first three columns) and landmark position (last three columns) Kalman filter gain values are equivalent in magnitude. (Ignore the zero-valued velocity state gains as off-diagonal cross-correlations were not present in this case, $\mathbf{P}$=Identity). Contrast this case with one in which the landmark position is 10 times more certain:

$$\mathbf{K} = \begin{bmatrix} 0 & .12 & 0 & 0 & 0 & 0 & -.12 & 0 & .25 & 0 & -.0012 & 0 \\ .031 & 0 & -.062 & 0 & 0 & 0 & 0 & .16 & 0 & -.0003 & 0 & .0006 \end{bmatrix}^{\mathbf{T}}$$

In this case, the landmark position gains are $1/100^{th}$ that of aircraft position. Nearly all of the aiding is distributed to the navigation states, and very little to the landmark position states. In the known landmark position tracking case, all aiding is distributed to the navigation states and none to the landmark position. As proposed above, as $\mathbf{P}_{survey} \rightarrow \mathbf{0}_{3x3}$, the effective state dimension decreases as the landmark position states essentially fall out. This examination of the Kalman filter gain indicates that no aiding action is lost in this pursuit (only the aiding action to the landmark position states is affected). This suggests that there exists a threshold below which, a well surveyed landmark will cause the EKF to behave (practically) like it would be for a landmark position that is known. This is true despite the fact that the state vector (having been augmented to estimate landmark positions) dimension exceeds the rank of the observability grammian. Due to the stationary nature of landmark positions, once below this threshold, this behavior persists as long as it is tracked. This behavior is exhibited for all valid realizations of $\mathbf{P}$.

It is thus proposed that, once a well surveyed landmark is injected, the algorithm performs as if the position is known as long as it's tracked. From an observability standpoint, this aiding benefit must be examined. Consider the case in which two unknown landmarks are being tracked. The augmented measurement equation when injecting the surveyed position for both landmarks is

$$\mathbf{H}'\left(\mathbf{x}_c(t)\right)=\begin{bmatrix}\mathbf{H}\left(\mathbf{x}_c(t)\right)\\ \left[\mathbf{0}_{6x9} \quad \mathrm{I}_{6x6}\right]\end{bmatrix} \tag{3.104}$$

where $\mathbf{0}_{6x9}$ is a matrix of zeros, and I_{6x6} is an identity matrix allowing the direct measurement of the last 6 states, the landmark position estimates. In this case, full rank is achieved. In fact, if the number of estimated landmarks is greater than 2, full rank is still achieved by measuring the position of just two. This implies that, despite many estimated landmarks, one need only know the location of two, to have complete observability. Section 3.3.1 showed this to be true in the known landmark case, but the extension to the estimated landmark position case is notable, and will prove fruitful in the next section. Granted, this measurement equation only applies at the epoch that the landmark position data is injected, but because the uncertainty is small (below an appropriate threshold), and the survey source is independent, the benefit is felt with every measurement afterwards.

A direct landmark position measurement is not taken at every epoch, but the effective state dimension is reduced, and the system becomes, for all practical purposes,

fully observable. It can thus be proposed that no direct measurement need be taken at the epoch of insertion either. One need only insert the landmark's independently *well* surveyed position and associated uncertainty covariance into the EKF as described above. This allows the seamless inclusion of surveyed landmarks into an algorithm designed to track and estimate LOO. No modification to the algorithm is necessary. The only difference between a LOO and surveyed landmark is the source and confidence of its position estimate. The SLAAMR algorithm leverages this finding to allow a single process to include both types of landmarks seamlessly without special handling functions.

It is purposed that, when using an EKF, injecting independently surveyed landmarks into a state vector comprised of estimated landmark position states will exhibit comparable performance to an algorithm that assumes perfect knowledge of landmark position (and does not augment the state vector for estimation). This is not to say that the assumption of perfect knowledge implies that landmark positions are actually perfectly known. No survey is perfect, and uncertainty is inherent. This proposed SLAAMR strategy accounts for imperfect surveys, and may actually perform better. A perfectly known landmark position also does not grant immunity to poor GDOP, as real cameras have finite resolution. This proposal will be evaluated experimentally in Chapter 4.

3.4 Self-Surveying Landmarks of Opportunity

In order to track a LOO, it must be inserted into the EKF. This is done in the same manner as the surveyed landmarks, as discussed above. The only difference is the source of the position estimate and the associated uncertainty. Through the course of

tracking a LOO, the uncertainty in its position estimate shrinks, but will never be less than that of the host aircraft. The SLAM concept asserts that, through bearings-only measurements, a 3-dimensional map of the surroundings can be created. Many techniques have been purposed, such as batch estimators and binocular vision. The SLAAMR algorithm uses the behavior described here. By simply inserting LOO into the EKF, the process of continual observation, or tracking, will allow the landmark estimates to shrink in uncertainty, and build a map.

3.4.1 Building a Map. A landmark map is essentially a collection of landmark position estimates, feature descriptors to allow tracking, and associated position uncertainties. Suppose aircraft position and attitude are known precisely. In this case, the resultant map created by flying around a space (and tracking landmarks) would also be precise. This can be accomplished through additional GPS aiding of the INS (either tightly coupled or loosely coupled) [21]. One need only augment the measurement equation with direct measurements of the aircraft position states (loosely coupled example).

$$\mathbf{H}'\left(\mathbf{x}_c(t)\right) = \begin{bmatrix} \mathbf{H}\left(\mathbf{x}_c(t)\right) \\ \begin{bmatrix} \mathrm{I}_{3x3} & \mathbf{0}_{3x\mathbf{M}} \end{bmatrix} \end{bmatrix} \tag{3.105}$$

where $\mathbf{M}$ is the length of the whole error state. Tightly coupled GPS updates (requiring estimation and measurement of satellite vehicle pseudo-ranges) are well established, though slightly more complex. In either case, this type of procedure is well understood

and will not be discussed further. If the navigation state is constrained tightly enough, the landmarks in the map can approach the threshold where they would be considered well surveyed. The takeaway is that the simple augmentation of the measurement equation (to allow a constrained navigation state), enables the basic SLAAMR algorithm to build a precise map of surveyed landmarks. This seamless process can be turned on and off with no additional modification to the underlying algorithm. The SLAAMR algorithm can transition from building precise maps, to using them for aiding with the 'flip of a switch'.

3.4.2 Constraining Drift with Brute Force. Consider now, that some off-board aiding (like GPS) is not available. Previously it was shown that tracking two known landmarks gives full observability. It was also suggested that tracking two well surveyed landmarks gives the same practical benefit, despite the number of LOO otherwise being tracked and estimated. Suppose two surveyed landmarks are inserted into the EKF at the beginning of a flight (a valid alignment is assumed). It is not unreasonable to have a small set of surveyed landmarks at the base of operations. As the initial two landmarks are tracked over time, it was shown that the aircraft's position uncertainty will approach that of the surveyed landmarks. It was also shown that the position estimates of the remaining LOO will approach that of the aircraft, and thereby the initial surveyed landmarks (assuming favorable geometry). Eventually, the initial landmarks will leave the camera field of view and no longer be able to be tracked. However, the remaining LOO, with positional uncertainties that have approached that of the original surveyed landmarks, take over; only two are required. The cycle repeats for the LOO that take the

place of the surveyed landmarks that were lost. As it goes, the well surveyed landmarks constrain the navigation state, which is then used to self-survey new landmarks to take their place. Figure 3.12 illustrates this process.

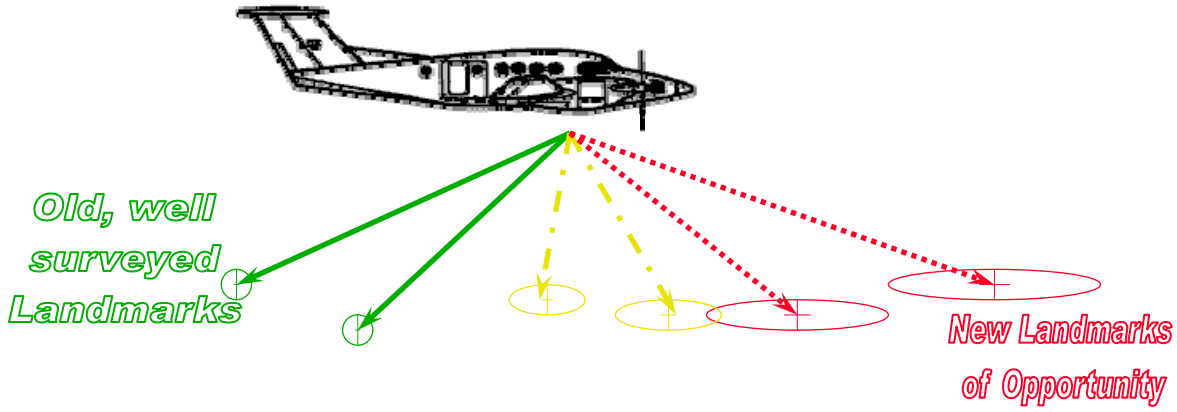

Figure 3.12: Iterative map building and aiding. New landmarks are tracked, refining their estimates and eventually used for primary aiding.

In essence, if the optics and inertial data were perfect, and the geometry just right, the system could continue indefinitely without drifting. The fact of the matter is, optics are flawed, an INS drifts, tracking is imperfect, and geometry is at the mercy of the flight path and geography.

The brute force approach (to constraining navigation drift) is to maximize likelihood of having good geometry and well surveyed landmarks by simultaneously tracking as many LOO, spread throughout the image, as computational power will allow. This approach statistically hedges the bet that there will always be at least two 'well surveyed features' available to limit drift. Realistically, errors in optics and tracking

guarantee a dilution of precision through this cycle, allowing drift to occur. The amount of drift is then a product of these factors. Chapter 4 will experimentally demonstrate that precise optics and this brute force approach can significantly reduce drift, without the aid of GPS or pre-surveyed maps. This is the key difference between the SLAAMR algorithm and previous work. SLAAMR is a single algorithm/process that can seamlessly transition from building precise maps with the aid of GPS, to using these maps to provide drift-free navigation, to navigating completely free of external help.

3.5 Establishing Initial Conditions

The equations from Section 3.2 provide the linearized EKF model for aiding an INS using the angular measurements obtained by tracking a single, stationary landmark. The extended Kalman filter assumes small perturbations about a true trajectory of the navigation-state variables. The true navigation state cannot be known for certain, so a calculated nominal (denoted by the c subscript), close to the truth, must be chosen. The calculated nominal, provided by the INS, is the truth plus the INS navigation error:

$$\mathbf{x}_{\mathbf{c}} = \mathbf{x}_{\mathbf{true}} + \delta\mathbf{x} \tag{3.106}$$

The question of how to choose the nominal trajectory is not trivial. The INS-derived nav-state provides a sufficient statistic for the first 9 states of the EKF nominal [15]. The initial estimate of the tracked landmark position requires some discussion. Large errors in the initial estimate invalidate the EKF linearization. Small errors are well

modeled by the linear approximation, and provide acceptable performance. Initial estimate errors must also exhibit a zero-mean, Gaussian nature in order for the EKF to perform as expected. The non-linear nature of optical measurements, and inability to deduce three-dimensional information from a single, two-dimensional image, make for a challenging problem. Binocular vision and multiple image batch estimation techniques have been developed and explored to provide initial object position estimates [5]. Hardware and other such constraints can make these approaches impractical. A number of monocular initial estimate strategies will be discussed and explored in Section 3.5.1. The position estimates are derived very differently, but take the form

$$\Delta_{tgt} = \mathbf{h}_{\Delta}\left(\mathbf{x}, \mathbf{z}\right)\Big|_{\mathbf{x_c}, \bar{\mathbf{z}}} \tag{3.107}$$

where $\mathbf{h}_{\Delta}$ is a non-linear function used to determine the landmark's relative position. The associated uncertainties take the form

$$\mathbf{P}_{tgt} = \mathbf{H_x}\mathbf{P_{xx}}\mathbf{H_x}^{\mathbf{T}} + \mathbf{H_z}\mathbf{P_{zz}}\mathbf{H_z}^{\mathbf{T}} + \mathbf{P_{pp}} \tag{3.108}$$

As described in Section 3.1.3.2, $\mathbf{H_x}$ and $\mathbf{H_z}$ are the partial derivatives of $\mathbf{h}_{\Delta}\left(\mathbf{x}, \mathbf{z}\right)$ with respect to $\mathbf{x}$ and $\mathbf{z}$, and $\mathbf{P_{pp}}$ is the aircraft position error co-variance. In this case, $\mathbf{P_{zz}}$ is defined in Section 3.2.2

$$\mathbf{P_{zz}} = \mathbf{H}[\tilde{\mathbf{x}}_{\mathbf{INS}}(t_i)] \cdot \mathbf{P_{xx}} \cdot \mathbf{H}[\tilde{\mathbf{x}}_{\mathbf{INS}}(t_i)]^{\mathbf{T}} + \mathbf{R} \tag{3.109}$$

where $\mathbf{P_{xx}}$ is error state uncertainty covariance matrix, $\mathbf{R}$ is the measurement noise covariance matrix, and $\mathbf{H}[\tilde{\mathbf{x}}_{\mathbf{INS}}(t_i)]$ is the observation matrix evaluated at the INS-derived nav-state.

3.5.1 Far Guess Landmark Initial Position Estimation. The first strategy to derive an initial position estimate involves very little a priori information about the environment. The position is estimated to be sufficiently far away and along the line-of-sight of the feature projection. The associated uncertainty ellipsoid maintains the tight angular uncertainty in the image, but assigns a large uncertainty along the LOS axis. Figure 3.13 illustrates this concept.

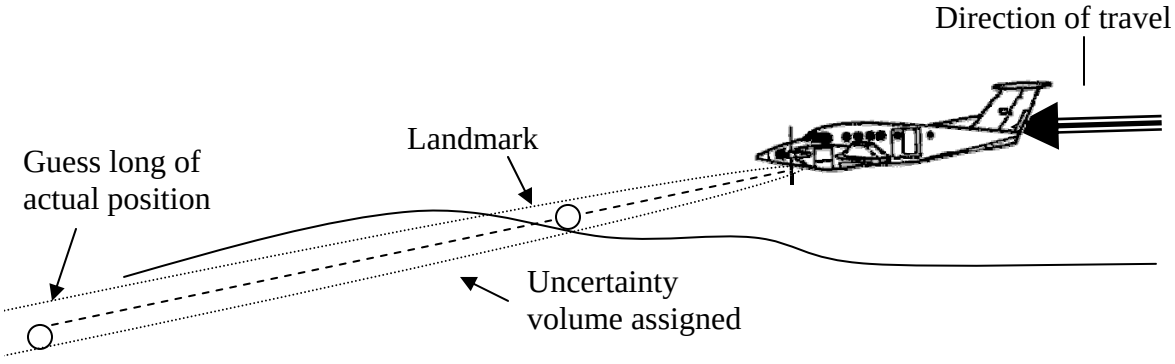

Figure 3.13: *Far Guess* initial landmark position estimation. The landmark position estimate is guessed long of the actual target, and uncertainty along the line of sight is assumed to be large

The relative position estimate is arrived at by multiplying the 3-dimentional unit vector feature projection ($\breve{\mathbf{z}}$) by a large distance (α):

$$\begin{aligned}\Delta_{tgt} &= \mathbf{h}_{\Delta}\left(\mathbf{x},\mathbf{z}\right)\Big|_{\mathbf{x_c},\bar{\mathbf{z}}} = \mathbf{T}_{ne}\cdot\left(\alpha\breve{\mathbf{z}}\right) \\ &= \mathbf{T}_{ne}\cdot\left(\alpha\mathbf{C}_c^n\frac{\mathbf{T}_{pix}^c\underline{\mathbf{z}}_p}{\left\|\mathbf{T}_{pix}^c\underline{\mathbf{z}}_p\right\|}\right)\end{aligned} \tag{3.110}$$

where

$$\breve{\mathbf{z}} = \mathbf{C}_c^n\frac{\mathbf{T}_{pix}^c\underline{\mathbf{z}}_p}{\left\|\mathbf{T}_{pix}^c\underline{\mathbf{z}}_p\right\|}, \qquad \underline{\mathbf{z}}_p = \begin{bmatrix} z_x & z_y & 1\end{bmatrix}^{\mathbf{T}} \tag{3.111}$$

$\underline{\mathbf{z}}_p$ is the homogeneous feature pixel location vector, $\mathbf{T}_{pix}^c$ and $\mathbf{T}_{ne}$ are transforms described in Chapter 2, and $\mathbf{C}_c^n$ is the camera-to-nav frame DCM.

The partial derivative of $\mathbf{h}_{\Delta}\left(\mathbf{x},\mathbf{z}\right)$ is then:

$$\mathbf{H}_{\mathbf{z}} = \mathbf{T}_{ne}\left(\alpha\mathbf{C}_c^n\mathbf{T}_{pix}^c\right)\begin{bmatrix}1 & 0\\ 0 & 1\\ 0 & 0\end{bmatrix} \tag{3.112}$$

Since the uncertainty along the LOS vector is ad hoc, α is chosen as the standard deviation for a camera frame z-axis uncertainty. This yields

$$\mathbf{H_x P_{xx} H_x^T} = \mathbf{T}_{ne}\mathbf{C}_c^n \begin{bmatrix} 0 & 0 & \alpha \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ \alpha \end{bmatrix} \mathbf{C}_n^c \mathbf{T}_{en} \tag{3.113}$$

Ideally, if zero knowledge was in fact the case, α would be equal to infinity, but machine precision, and matrix condition drive a real, but large number to be substituted.

An inherent problem with this method is that it does, by design, invoke a large error in the estimated landmark position, complicating the EKF linearization. The large error covariance assigned to the estimate aims to minimize the effect of this bad estimate. The large uncertainty opens the search for a match. When a match is found, the aiding derived from the measurement residuals is directed towards the dimensions with the highest uncertainty. This differs from a batch estimator in many ways. A batch estimator requires translation in order to triangulate the position of the landmark. Until enough apparent translation occurs, the batch estimator can provide no aiding. By injecting the LOO immediately and assigning a large uncertainty along the LOS, the EKF can reap the benefit of aiding along the dimensions that are certain. For example, a landmark on the distant horizon exhibits no angular rate and could therefore not be used by a batch estimator. The EKF would gain no positional aiding from such a landmark, but would gather attitude aiding in all dimensions accept rotation about the LOS vector. If slow translation is occurring, the EKF will gradually shrink the uncertainty in the dimensions that were un-observable, reaping ever more reward along the way. The batch estimator waits until translation has occurred before using the landmark. This technique is experimentally evaluated in Chapter 4.

3.5.2 Average Height Landmark Initial Position Estimation. The second strategy to derive an initial position estimate involves some a priori information about the environment. The *Average Height* method borrows from the *Far Guess* method. Instead of guessing that the landmark is far away, the guess is made that it is roughly co-planar with the others being tracked. Figure 3.14 illustrates this concept.

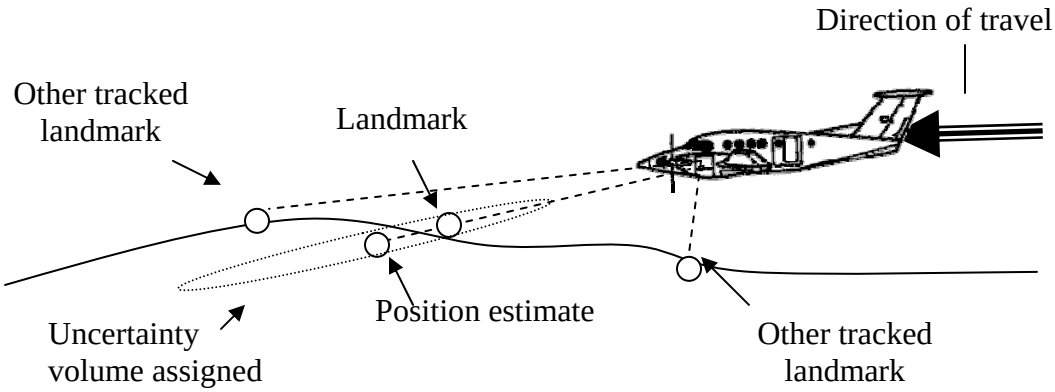

Figure 3.14: *Average Height* landmark position estimation. The landmark position estimate intersects the plane defined by the average height of other tracked landmarks.

The assumption is that landmarks close together have spatially correlated elevations. The implementation is identical to the *Far Guess*, except that the distance α is chosen such that the line defined by the unit pointing vector $\check{\mathbf{z}}$, intersects the x-y plane defined by the average height of all other tracks. The strength of the uncertainty along the LOS must be chosen based on knowledge of the environment, and is scaled by the inverse sin of the angle between the x-y plane and the LOS vector. The uncertainty can still be large, however. This is an engineering judgment call, not pure science by any

means. This technique does allow closer estimates over flat terrain, but is problematic in rising or rough terrain.

3.5.3 DTED Intersection Initial Landmark Position Estimation. The last strategy is again similar to the *Average Height*, except a DTED map is used for determining the height of the landmark, not the others being tracked. The unit pointing vector $\check{\mathbf{z}}$, is projected from the aircraft to the DTED map. An altitude and geodetic position are resolved recursively. This becomes the position estimate. Figure 3.15 illustrates this concept.

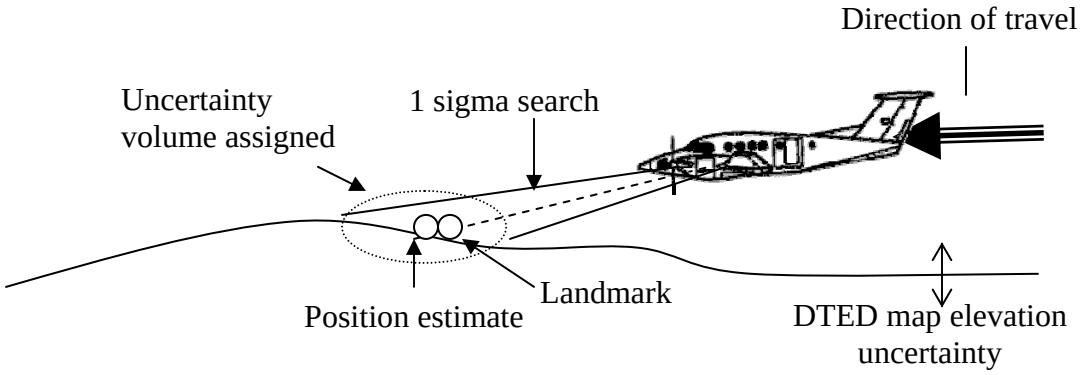

Figure 3.15: DTED intersection initial landmark position estimation. The landmark position estimate intersects DTED map.

The uncertainty in height becomes a function of the resolution of the DTED map itself, and the variance of terrain elevations in the immediate area ($\sigma^2_{\text{surroundings}}$). This variation comes from searching 1 standard deviation around the intersection point and

examining the terrain (according to the DTED map). Uncertainty in the position measurement is now a sum of the camera frame pointing errors, projected on the Earth, and the height uncertainty as described.

The relative position estimate is arrived at by intersecting the DTED map. The procedure to do so is trivial and not discussed. However, Δ_{tgt} is no longer a tractable function that can be differentiated, making the uncertainty derivation more ad hoc. The contribution to uncertainty due to the DTED map and aircraft navigation state can be modeled as the sum of DTED uncertainty and the variation of the immediate surroundings.

$$\mathbf{H_x P_{xx} H_x}^{\mathsf{T}} = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^{\mathsf{T}} \left(\sigma^2_{\mathbf{DTED}} + \sigma^2_{\mathbf{surroundings}} \right) \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \tag{3.114}$$

The remainder is still due to camera frame pointing errors. $\mathbf{P_{zz}}$ is still tractable, therefore

$$\mathbf{H_z} = \mathbf{T}_{ne} \left(\alpha_{\mathbf{DTED}} \mathbf{C}_c^n \mathbf{T}_{pix}^c \right) \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} \tag{3.115}$$

where, $\alpha_{\mathbf{DTED}}$ is the length of the relative position vector to the DTED intersection

$$\alpha_{\mathbf{DTED}} = \left\| \Delta_{tgt} \right\| = \left\| \mathbf{T}_{en} \cdot \left(\mathbf{p_{ac}} - \mathbf{p_{landmark}} \right) \right\| \tag{3.116}$$

where $\mathbf{p}_{\mathbf{ac},\mathbf{landmark}}$ are the geodetic positions of the aircraft and landmark.

This technique gives excellent position estimates as long as the navigation state is close to truth. If the position or attitude errors are too large, the intersection will become invalid. This is by no means an independent measurement source.

3.5.4 Replacing Landmark Tracks. The SLAAMR routine establishes a navigation state augmented with a constant number of estimated landmark tracks. This is as opposed to dynamically building the state vector based on the number of features available in each image. This method was chosen for two reasons. SIFT© consistently generates hundreds of features or more, so it is impractical to track all. It was also shown that only 2 are ever needed, but many should be tracked to ensure good aiding action. Ten to twenty tracks (constant for a given scenario) is the chosen compromise. When SLAAMR initializes or a landmark falls out of view, a new landmark must be inserted in its place. A new landmark is chosen and its position estimated by one of the methods described above. Its position estimate ($\mathbf{p}_{\mathbf{new}}$) is simply added in the vacant spot in the whole nave state:

$$\hat{\mathbf{x}}(t_i) = \begin{bmatrix} \hat{\mathbf{p}}_{ac}^{\mathsf{T}} & \hat{\mathbf{v}}_{eb}^{n\,\mathsf{T}} & \hat{\Psi}_{nb}^{n\,\mathsf{T}} & \vdots & \hat{\mathbf{p}}_{tgt1}^{\mathsf{T}} & \cdots & \mathbf{p}_{\mathbf{new}}^{\mathsf{T}} & \cdots & \hat{\mathbf{p}}_{tgt\mathbf{N}}^{\mathsf{T}} \end{bmatrix}^{\mathsf{T}} \tag{3.117}$$

and the associated error state components are set to zero:

$$\delta\mathbf{x}(t_i) = \begin{bmatrix} \delta\mathbf{p}_{ac}^{\mathsf{T}} & \delta\mathbf{v}_{eb}^{n\,\mathsf{T}} & \mathbf{e}_{nb}^{n\,\mathsf{T}} & \vdots & \delta\mathbf{p}_{tgt1}^{\mathsf{T}} & \cdots & \begin{bmatrix} 0 & 0 & 0 \end{bmatrix} & \cdots & \hat{\mathbf{p}}_{tgt\mathbf{N}}^{\mathsf{T}} \end{bmatrix}^{\mathsf{T}} \tag{3.118}$$

The associated 3-dimensional uncertainty ($\mathbf{P}_{new}$) is inserted into the error state covariance matrix ($\mathbf{P}(t_i)$), with zeros filling the cross covariance elements.

$$\mathbf{P}(t_i) = \begin{bmatrix} \mathbf{P}(t_i)_{1:j,1:j} & \mathbf{0}_{j\text{x}3} & \mathbf{P}(t_i)_{1:j,(N-j+4):N} \\ \mathbf{0}_{3\text{x}j} & \mathbf{P}_{new} & \mathbf{0}_{3\text{x}(N-j+4)} \\ \mathbf{P}(t_i)_{(N-j+4):N,1:j} & \mathbf{0}_{(N-j+4)\text{x}3} & \mathbf{P}(t_i)_{j+4:N,j+4:N} \end{bmatrix} \tag{3.119}$$

as described in Section 3.1. The zeros filling the cross-covariance elements ensure that the new track is independent of the other states until it is tracked for the first time. This allows a seamless, online method of inserting new tracks without disrupting the others.

3.6 Principles of Design

This chapter developed the theory behind the design of the SLAAMR algorithm. A number of principles ultimately drive trade-offs and strategies in an effort to maximize aiding in a robust environment. They are:

- Only two known landmark tracks are required for full observability.
- Aiding action increases with dwell time.
- Surveyed landmarks (though estimated in the EKF) provide nearly equivalent aiding to known landmark tracking.
- Through continuous tracking, the uncertainty of landmark and aircraft position estimates approaches that of the lower one.
- Track many landmarks simultaneously to dilute unobservability.
- Choose new landmarks to track, such that the population evenly spans the image.
- Include landmarks into the EKF immediately (after a single image) to maximize aiding (don't wait until triangulation is possible, i.e., as in batch processing).
- Use a wide field-of-view lens to maximize dwell time.
- Point the camera down or backwards to maximize good geometry and dwell.
- Do not point the camera forward as it provides poor temporal feature flow.
- Passive altitude and heading measurements improve overall observability.
- Low *V over h* generally improves the observability grammian matrix condition number and GDOP, but too low has penalties.
- There exist *sweet spots* at higher *V over h* that improve both the observability grammian matrix condition number and GDOP, but cannot be predicted.
- High and low values of *V over h* are relative and cannot be generalized. Without knowledge of these unknown thresholds, the *V over h* span should be maximized to capture the sweet spots and complementary ends of the spectrum.
- Capturing the horizon provides attitude observability.
- Capturing nadir improves possibility for good GDOP.

3.7 SLAAMR Algorithm Design

The SLAAMR algorithm was based on the study done in this chapter. It is designed to maximize aiding strength at all times in the worst case navigation scenarios. The addition of passive measurements, such as heading and altitude, complement the SLAM based design to make up for its observability lackings. Additional design elements that account for realistic image processing are implemented as well. Figure 3.16 illustrates the SLAAMR algorithm process with a block diagram.

The INS and EKF act as a conventional feedback system, where error state values are applied to the INS nav-state with every measurement. The same process applies to the landmark position estimates. The compass and altimeter measurements are added to the system by simply augmenting the EKF measurement equation.

The camera is pointed to optimize feature flow (down or aft). SIFT© algorithm creates features from images. These features are compared to tracked landmarks to find a match. Estimated landmark positions are used to narrow the search. If a new landmark track is desired, a feature is selected from the image based on its location in the frame and relation to the other tracks. Then an initial position estimate is made by one of the various techniques discussed, and the landmark inserted into the system. If surveyed landmarks are available, the features are searched for a match. If a match is made, the surveyed landmark is inserted, overriding a landmark of opportunity track.

The estimated measurements (feature projections, heading and altitude) are generated by the navigation state and landmark position estimates, and the differences with measured values are fed into the EKF as residual errors.

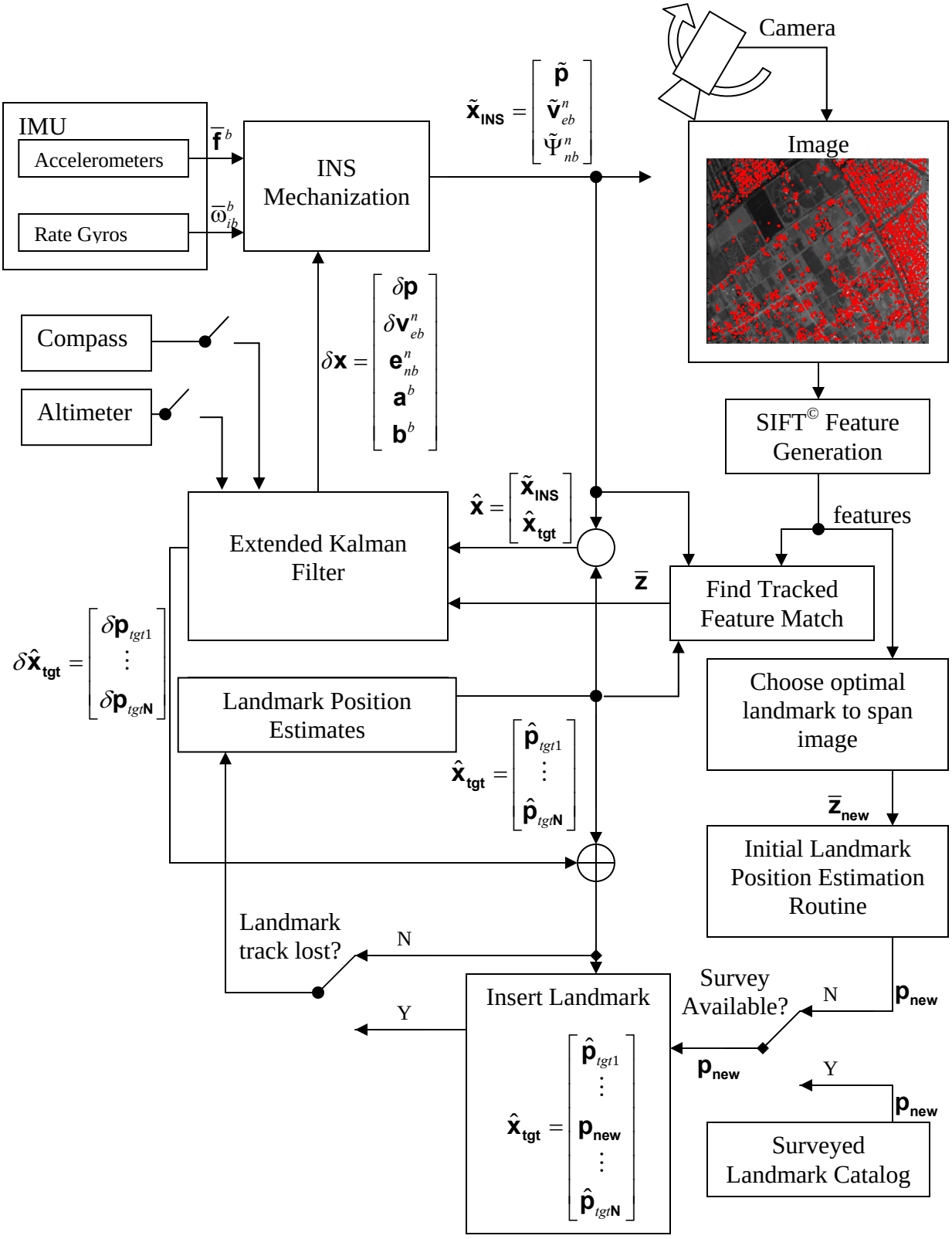

Figure 3.16: SLAAMR System Diagram.

3.8 Design Summary

This chapter aimed to explore the mathematical nature of bearings-only measurements and how they apply to a SLAM-based navigation strategy. The findings illustrated several influencing factors and challenges that hinder that performance of an EKF aiding strategy. A mathematically optimal design was augmented with practical measures to overcome realistic problems faced when designing a robust system capable of operating in all environments.

The importance of camera look angle and a few critical additional passive measurements make the design of the SLAAMR algorithm stand apart from most SLAM-based systems, and allow it to perform well in realistic environments with no artificialities. The SLAAMR design does not rely on perfect and non-existent image processing, orbiting a set of favorable landmarks, or the benefits of small scale.

The next chapter will experimentally evaluate the mathematical assertions made in this chapter. Experimental data will also support the design decisions made by illustrating their benefit and the penalties paid for not being implemented.

IV Flight Test and Experimental Results

Chapter 3 established the mathematics behind the SLAAMR routine and established the basic architecture. During the mathematical development, some propositions were made about the performance of the SLAAMR algorithm and factors influencing aiding action. This chapter will experimentally evaluate both the SLAAMR algorithm, and underlying principles behind image aiding of an INS.

The flight test equipment and methods are described, as well as limiting factors. A precision truth source is used to validate the results, providing clear data and conclusions. Navigation state errors and the statistics of these errors are used as appropriate measures of performance for the navigation system.

The experimental portion of this thesis was flown as a Test Management Project at the United State Air Force Test Pilot School (USAF TPS) in conjunction with the Air Force Institute of Technology (AFIT). The GPS Out-Or-Denied Low-Cost Kalman Iterated Navigation (GOOD LOOKIN) test team that executed the flight test was comprised of two test pilots, and three flight test engineers. Raw data were collected to evaluate the SLAAMR algorithm as a navigation strategy. A total of 13.1 hours on 6 flight test sorties were flown on a modified C-12C aircraft in the R-2508 complex and Palmdale/Lancaster, CA airspace between 6-18 September 2007. All experimental analysis in this chapter is derived from this test.

4.1 Simulation

Before any flight test was scheduled or flown, a simulation was performed to characterize performance. The SLAAMR algorithm was designed such that it could be fed simulated images and inertial measurements in the same manner as it would with real data. This allowed the algorithm to go through many simulated iterations and changes before flight, with a high confidence that the flight test algorithm would function in a manner similar to simulation. Performance may vary with real data inputs, but the functionality of the algorithm should be the same.

Each profile or mode of data collection was simulated before flight. Simulated results validated the overall design of the SLAAMR algorithm. The only difference between flight test performance and simulated performance would reside in the nature of the measurement data fed into the algorithm. The algorithm attempts to model the nature and noise associated with the inertial and image measurements, and thus, degraded performance in flight test is likely due to these models being insufficient. Simulated results will accompany flight test results, presented later in this chapter.

4.1.1 Simulated Images. The simulated images came in the form of feature sets (or key files), identical in format to those produced by the SIFT© algorithm. A set of features was generated in 3-*dimensional* space, and assigned geodetic coordinates. Each feature was assigned a scale, orientation and 128-element descriptor using a random number generator. A true aircraft trajectory was created, simulating the aircraft's position, velocity and attitude in space. Key files were generated at appropriate time intervals ($t=t_i$), using the true aircraft trajectory, and true landmark positions ($\mathbf{x}_{\mathbf{true}}(t_i)$).

The true pixel coordinates in each key file ($\mathbf{z}_{\mathbf{true}}$) were generated using the non-linear measurement equation from Chapter 2:

$$\mathbf{z}_{\mathbf{true}}(t_i) = \mathbf{h}[\mathbf{x}_{\mathbf{true}}(t_i)] \tag{4.1}$$

The true pixel coordinates were generated according to a pinhole camera model. Wide angle, fish eye distortion was applied to these coordinates using a distortion function ($D(\bullet)$), derived in Section 4.3.2.

$$\mathbf{z}_{SIM}(t) = D\left(\mathbf{z}_{\mathbf{true}}(t_i)\right) + \mathbf{v}_{\mathbf{m}} \tag{4.2}$$

White Gaussian pixel coordinate measurement noise ($\mathbf{v}_{\mathbf{m}}$) was added according to the model in Chapter 2. The simulated pixel coordinates $\mathbf{z}_{SIM}(t)$ differ from true measurements in that the noise is assumed to be zero-mean and normally distributed. The test data are generated by, and thus corrupted by, the feature generation program and non-linear optics. Additionally, white noise was added to the descriptors to simulate change frame to frame. This is by no means the true nature of feature descriptor change, which proved problematic, as will be shown.

4.1.2 Simulated Inertial Measurements. The true trajectory (generated for the simulated flight path) was used to generate simulated inertial measurements. The equation derived in Chapter 2 is solved for the specific force vector $\mathbf{f}_{ib}^{b}$:

$$\mathbf{f}_{ib}^{b} = \mathbf{C}_{n}^{b}\left(\dot{\mathbf{v}}_{en}^{n} + (2\boldsymbol{\omega}_{ie}^{n} + \boldsymbol{\omega}_{en}^{n}) \times \mathbf{v}_{en}^{n} - \mathbf{g}_{l}^{n}\right) + \mathbf{a}^{b} + \mathbf{w}_{a}^{b} \tag{4.3}$$

and is a function of aircraft nav-to-Earth velocity and acceleration ($\mathbf{v}_{en}^{n}, \dot{\mathbf{v}}_{en}^{n}$) in the *e-frame*, the accelerometer bias ($\mathbf{a}^{b}$ modeled as a first-order Gauss-Markov process), and a number of Earth effects. The same was accomplished for the rate gyro measurements $\boldsymbol{\omega}_{ib}^{b}$:

$$\boldsymbol{\omega}_{ib}^{b} = \boldsymbol{\omega}_{nb}^{b} + \mathbf{C}_{n}^{b}\left(\boldsymbol{\omega}_{ie}^{n} + \boldsymbol{\omega}_{en}^{n}\right) + \mathbf{w}_{b}^{b} \tag{4.4}$$

which is a function of the attitude, body-to-nav rotation in the *b-frame* ($\mathbf{C}_{n}^{b}, \boldsymbol{\omega}_{nb}^{b}$) and Earth effects.

The simulated inertial measurements were corrupted with white, Gaussian noise ($\mathbf{w}_{a}^{b}$, $\mathbf{w}_{b}^{b}$) according to the model in Chapter 2. They differ very little from the flight test data because these kinematics and models are very well developed and understood [23]. True sensors, however, are subject to aircraft vibrations. This was not simulated.

4.2 Test Item Description

The system under test (SUT) consisted of two high-resolution digital still cameras, a micro-electro-mechanical system (MEMS) inertial measurement unit (IMU), and a laptop computer for data collection and processing. The components were selected to provide a low-cost, precision navigation system for use in GPS-denied environments. The SUT was carried aboard a C-12C aircraft, allowing an unobstructed view of the Earth below. Precision Time Space Position Information (TSPI) was provided by the 412th Range Squadron, and correlated to the test data.

4.2.1 High Resolution Digital Camera. The Pixelink PL-A741 is a high-performance, 1.3 mega-pixel monochrome camera designed specifically for machine vision applications. The camera is connected to the computer by a FireWire interface [33]. It provided the images needed for aiding, and an external trigger option enabled GPS time stamping of the images, which is critical for effective aiding using the SLAAMR algorithm.

The PL-A741 had a selectable resolution, but 1024x768 was used for test. The shutter, gain, and contrast were selectable prior to data runs and were set for environmental conditions. The cameras generated Portable Gray Map (.pgm) images, which are free from compression distortions. The camera was triggered to take images at a rate of 4 per second. A higher rate was desired, but as it stood, each image was 784 kilo-bytes, and approximately 50-60 giga-bytes of data were collected per hour (two cameras collecting simultaneously).

The lens used provided a field-of-view of 63.2 degrees vertically and 84.7 degrees horizontally. The cameras were arranged such that the wider dimension ran parallel to flight path, maximizing longitudinal field of view. The focal length of the lens was 4.9268 mm, and the lens set to focus on infinity. A summary of camera specifications follows. The focal length (f), focal plane pixel dimension (MxN) and physical height (H) and width (W) (all used in Equation (3.30)) were:

$$f = 4.9268(mm) \tag{4.5}$$

$$M = 768, \qquad N = 1024 \tag{4.6}$$

$$H = 5.4979(mm) \qquad W = 7.4044(mm) \tag{4.7}$$

Figure 4.1 illustrates the compact camera package. Further specs can be found in reference [33].

Figure 4.1: High resolution digital camera – PL-A471.

4.2.2 Inertial Measurement Unit. The Microbotics INC., MIDG II INS/GPS Inertial Navigation System (INS) was chosen as the Inertial Measurement Unit (IMU) for this test. Though, marketed as a GPS/INS, it was in fact used simply as an IMU for this test. The raw inertial measurement data (from the internal accelerometer and rate gyro triads) was collected at 50 samples per second, via a serial cable. The internal inertial sensors were Micro-Electro-Mechanical Systems (MEMS) level components. The onboard GPS receiver (connected to an external antenna) allowed GPS time stamping of the inertial data as well as the images. The GPS data (collected separately from the inertial measurements) also served as a redundant truth source in the event that the TSPI failed. The MIDG II was chosen because it was representative of a MEMS grade sensor, while providing a convenient interface and GPS time stamp capability.

The values for accelerometer measurement noise ($\mathbf{w}_a^b$), bias process driving noise ($\mathbf{w}_{a_{bias}}^b$), and time constant (τ_a) were derived from ground test data and estimated to be Gaussian with strengths defined by:

$$\mathbf{w}_a^b \rightarrow \mathbf{Q}_a = \left(0.050\, m/s^2\right)^2 \cdot s \tag{4.8}$$

$$\mathbf{w}_{a_{bias}}^b \rightarrow \mathbf{Q}_{a_{bias}} = \left(0.20\, m/s^3\right)^2 \cdot s \tag{4.9}$$

$$\tau_a = 1/3600 \left(\text{sec}\right) \tag{4.10}$$

The values for rate gyro measurement noise ($\mathbf{w}_b^b$), bias process driving noise ($\mathbf{w}_{b_{bias}}^b$) and time constant (τ_b) were estimated to be:

$$\mathbf{w}_b^b \rightarrow \mathbf{Q}_b = \left(0.060\, rad/s\right)^2 \cdot s \tag{4.11}$$

$$\mathbf{w}_{b_{bias}}^b \rightarrow \mathbf{Q}_{b_{bias}} = \left(0.0087\, rad/s^2\right)^2 \cdot s \tag{4.12}$$

$$\tau_b = 1/3600 \left(\text{sec}\right) \tag{4.13}$$

The MIDG II was small (1.50" x 1.58" x 0.88", weight under 55 grams), and illustrated in Figure 4.2. More information can be found in [34].

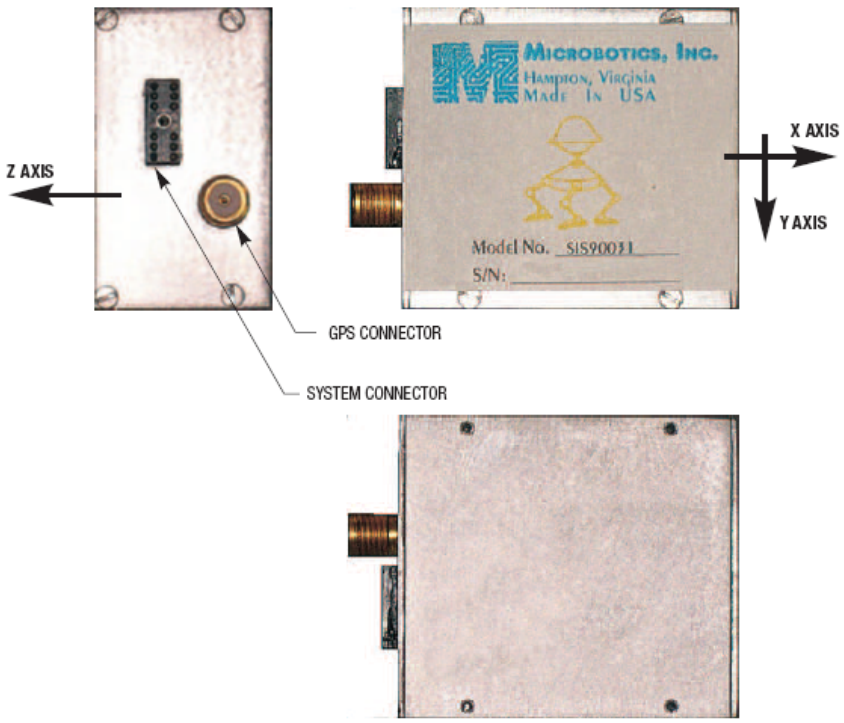

Figure 4.2: Inertial Measurement Unit – MIDG II INS/GPS.

4.2.3 Truth Reference Source. A combined GPS/inertial reference system and external GPS antennas provided truth data for post flight performance assessment. The truth reference system used (GLite Configuration 2-B, C2B) integrates the Novatel OEM-V GPS receiver, the Honeywell Ring-Laser-Gyro HG1700AG58 IMU, and various other Commercial Off-The-Shelf products to record GPS/IMU data, produce a GPS/IMU blended solution in real-time, and interface with various monitors and controls devices. The data was recorded via removable media and post-processed by the 412th Range Squadron at Edwards AFB. The resultant truth data provided precise (differential GPS quality) geodetic position, velocity and attitude data. These data correspond to the nine basic nav-states output by the INS under test. The GLite was positioned close to the MIDG II (2 meters behind). A simple three axis rotation was required to align the MIDG II frame of reference and the GLite. A precise survey of the GLite frame of reference, aircraft frame, and MIDG II frame was performed by Air Force Flight Test Center (AFFTC) personnel in order to resolve this alignment. Note that the aircraft frame was a relatively arbitrary intermediate frame between the GLite and MIDGII, and is thus discarded. The MIDG II frame will be considered to be the aircraft frame for the purpose of this discussion. The precision and accuracy of this system is orders of magnitude better than the system under test will achieve, and is therefore considered a practical truth.

4.2.4 Sensor Integration. The SUT consisted of two basic sensors: the cameras and the IMU. It also used two sources of truth data, the TSPI and GPS data from the MIDG II. All the data was collected into a laptop computer (XPS M1710 Special Edition

Formula Red) carried onboard the aircraft. A data collection program (created at AFIT by Veth for his work on the subject [26]) performed the job of collecting, sorting, and controlling the sensors. All measurements and images were stamped with GPS time to allow proper time synchronization during post processing. The TSPI data and inertial data were taken at a rate of 50 samples per second and simultaneously on the $1/50^{th}$ of a GPS week second (i.e., data taken at 0.00, 0.02, 0.04…). Though not required, this reduced the effort required to correlate the truth with estimates.

Two cameras were used at a time, each pointing at a different angle. One was declared master and one slave. An internal trigger in the master camera fired (at a rate of 4 per second), commanding the slave also to fire. The trigger notified the data collection program that images were being taken, and the GPS time gathered from the MIDG II was them 'stamped' on the images in the form of the image's filename. Figure 4.3 illustrates a top-level block diagram of the SUT.

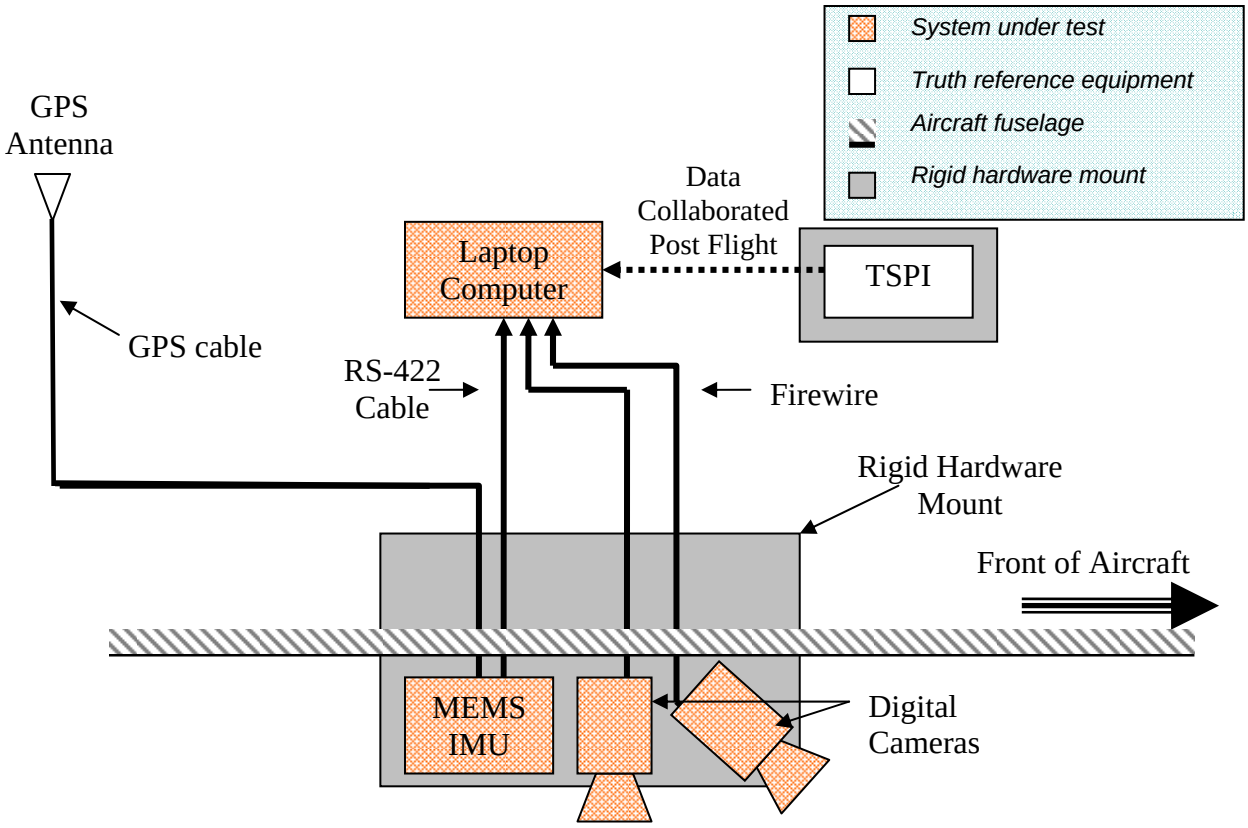

Figure 4.3: System under test block diagram.

4.2.5 Test Aircraft. A C-12C Huron aircraft (shown in Figure 4.4) was chosen for the test. The Huron is a twin engine turbo-prop, utility class aircraft. Its performance and operating envelope is comparable to many transport aircraft or the predator UAV.

Figure 4.4: Test aircraft, C-12C Huron.

The Huron allowed the entire test team to be onboard during test. The team comprised of 2 test pilots, 1 mission director, and 1-2 systems operators. Figure 4.5 shows the crew positions and equipment racks.

The GLite is just out of sight in Figure 4.5, on the floor next to the orange equipment rack. The cameras and IMU are below the co-pilot's seat (not in view) and external to the aircraft. There existed a hatch below the co-pilot's seat from a previous test, allowing the seamless installation of the cameras and IMU. The cameras and IMU were rigidly mounted to an existing apparatus that fits into this hatch (described in the next section), and wires run under the floor boards to the computer. The cameras were individually configurable to allow them to be positioned within 45 degrees of vertical,

along the pitch axis. Unnecessary antennas were removed from the bottom of the aircraft to allow an unobstructed view from horizon to horizon. Figure 4.6 shows the cameras as installed in the aircraft.

Figure 4.5: Crew positions and test equipment.

Figure 4.6: Installed Sensors.

The left propeller is seen in the background of Figure 4.6. The front of the aircraft is to the right of the image. In this figure, one camera is looking forward, and one down, but they could be repositioned to meet the needs of the test mission.

The C-12C provided a stable platform for flight test. The turbo props provided a relatively low vibration environment. The test parameters were well within the operational limits of the aircraft, and the designed structural limits of the camera harness were well beyond those of the aircraft. The C-12C aircraft was not a limiting factor for test, nor was there any danger of damaging the equipment if in the C-12C flight envelope.

4.2.6 Sensor Harness. As mentioned above, the cameras and IMU were rigidly mounted to a sensor harness that mated to the pre-existing hatch below the co-pilot's seat. The harness slid down rails and latched, mating with the lower fuselage, and forming a

pressure seal. The harness was further modified to mount the MIDG II IMU and cameras rigidly and close together. The distance was small enough to be neglected in computation (<1-2 inches). Figure 4.7 show the harness in more detail.

The IMU was roughly aligned with the natural aircraft body axis, and thus the body frame was defined to be equal to the IMU frame. The cameras were mounted coplanar with the IMU, and set in pivoting grooves to allow a +-45 degree look angle change. Once set at the desired look angle, the cameras could be securely fastened. One camera was aligned with the IMU, looking straight down, and locked into place. This camera's frame of reference and the IMU frame were precisely surveyed and never changed. The remaining camera was set to different look angles according to test requirements (roughly + or – 45 degrees), and its rotation angle was measured relative to the fixed, downward camera. Since the cameras were coplanar, only the pivot angle needed to be measured.

The harness was designed to withstand the dynamic air pressure created by flying at 250 knots indicated airspeed with a margin of safety of 1.6. It was flight tested to 220 knots without deformation or vibration.

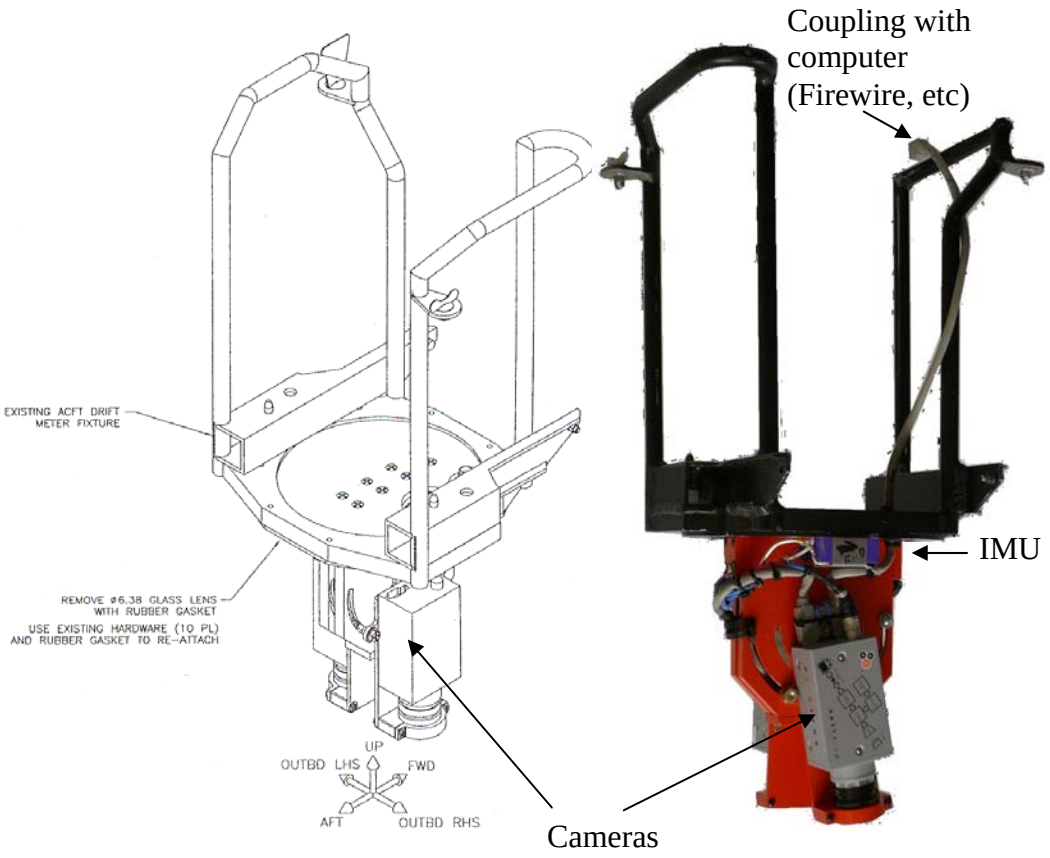

Figure 4.7: Sensor harness.

4.3 Test Item Calibration and Validation

The underlying theory behind aiding a low-cost INS with bearings measurements, assumes that certain parameters are known or well modeled. Bearings measurements from cameras require high image photo resolution for precision and a true angular survey for accuracy (called a boresight). The image resolution not only applies to the resolution of the image itself, but accounts for the feature generation process, and in part, the pinhole camera model calibration used. The angular survey requires precise knowledge of camera alignment to the IMU frame (boresight), field of view and pinhole camera model calibration.

The Kalman filter also assumes certain properties of the feature generation system. The EKF model performs best when features are tracked until no longer in view (to maximize dwell time). The ability of the feature generator to place a feature consistently on a landmark drives performance and process efficiency. The nature of the feature generator must be well understood and the algorithm optimized for accuracy, precision, and persistence.

This section discusses the process used and results from the calibration of the test equipment and feature generation software. Ground-based techniques as well as precise truth data from flight test are used to derive and validate the test equipment calibration and characterization.

4.3.1 Camera Field-of-View. The field-of-view of the camera is required to deduce the effective size of the imaging array. This measurement is required for the

pinhole camera model. Width (W) and Height (H) are defined by function of the horizontal and vertical fields of view (FOV_h , FOV_v) .

$$W = 2 \cdot f \cdot \tan\left(\frac{FOV_h}{2}\right), \qquad H = 2 \cdot f \cdot \tan\left(\frac{FOV_v}{2}\right) \tag{4.14}$$

The ultimate goal of calibration was to reduce errors to less than 1 pixel. A point source object (corner of a checkerboard panel) was placed far from the camera and such that it was projected near the edge of the camera field of view. The distance and angles from the camera focal point and the object were precisely measured, and field of view determined with trivial geometry. This was repeated on both sides, and in both axes to find asymmetry. The center of the field of view must also be known, and expressed as a pixel location to define the z-axis of the *c-frame*. Since physical surveys were made relative to the physical camera shell, so must the camera frame projection in the image. Lasers were aligned to the shell on both sides, and swept. The intersection of the laser traces on a far away board (as seen in the image), defined the center pixel. This was compared with the results of the pinhole camera calibration process defined in [36].

4.3.2 Pinhole Camera Model Calibration and Results. The first objective of this test was to determine pinhole camera calibration error model for the wide angle digital cameras. The lens of the camera distorted the image in a non-linear fashion. The wide field of view (FOV) caused a “fish eye” effect, producing an image that compresses features near the edge. This distortion was inconsistent with the pinhole camera

approximation used in the measurement equation. The fish-eye effect had to be removed to approximate a pinhole camera. The distortion was modeled as a sixth-order polynomial function of radial distance (from the center of the image). An un-distort transform ($U(\bullet)$) and its inverse (the distort transform $D(\bullet)$) were created to convert image coordinates between real, measured and distorted coordinates ($\bar{\mathbf{z}}_d$) and pinhole camera image space containing undistorted, calculated coordinates ($\tilde{\mathbf{z}}_u$).

$$\tilde{\mathbf{z}}_u = U(\bar{\mathbf{z}}_d) = 1 + \sum_{i=1}^{6} a_i \left\| \bar{\mathbf{z}}_d \right\|^i$$

$$D(\bullet) = U(\bullet)^{-1}$$

$$\tilde{\mathbf{z}}_d = D(\tilde{\mathbf{z}}_u)$$

The function $D(\bullet)$ relies on iteration, thus a small error can be induced by repeating the distort/un-distort process back and forth ($\tilde{\mathbf{z}}_d \neq \bar{\mathbf{z}}_d$). For the purpose of this evaluation, the error is negligible for single transformations.

A discussion of the process to derive this model and transform is contained in [36]. It is a well understood process, and not original work in this thesis. Thus this process will not be discussed in detail. However, Figure 4.8 shows the result of the distortion removal calibration.

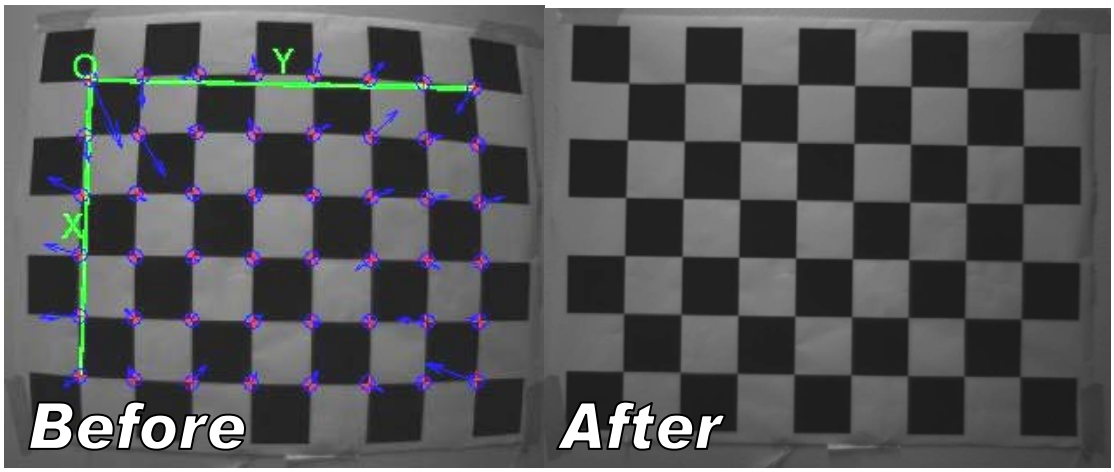

Figure 4.8: The effects of the pinhole camera calibration and transform.

To speed up the SLAAMR algorithm, the pinhole camera calibration model was not applied to the actual images, but simply applied to all feature image locations generated by the SIFT© algorithm. The resultant feature locations were then expressed in the focal plane of a pinhole camera. This method also prevents feature distortion and aliasing that may be a result of transforming the whole image, and then running SIFT©.

In order to determine the camera calibration error model, truth camera position, attitude data (derived from TSPI), and surveyed landmark positions were used to correlate predicted and measured landmark image projections. The difference between estimated and measured image projection was considered error. Errors were measured across the image plane and a calibration error model was generated. The errors were characterized by mean and standard deviation, as well as spatial dependence across the image.

Precise surveyed landmark data were obtained by over-flying precisely surveyed landmarks on the Edwards Air Force Base Photo Resolution Road (PRR). The PRR is populated with targets of varying sizes and shapes that are correlated to precise geodetic coordinates. The non-linear measurement equation, evaluated for the truth data

$$\tilde{\mathbf{z}}_u = \mathbf{h}[\mathbf{x}_{\mathbf{TSPI}}] \tag{4.15}$$

generates the predicted image projection in the pinhole camera image space.

The actual landmark was manually tracked frame-to-frame, generating measurements in distorted, real image space ($\bar{\mathbf{z}}_d$). The predicted image projections were transformed into real image space and differenced with the predictions. The result was a set of error vectors associated with portions of the image.

$$\begin{aligned} \delta\mathbf{z}_{err}(z_x, z_y) &= D\left(\tilde{\mathbf{z}}_u\right) - \bar{\mathbf{z}}_d \\ &= D\left(\mathbf{h}[\mathbf{x}_{\mathbf{TSPI}}]\right) - \bar{\mathbf{z}}_d \end{aligned} \tag{4.16}$$

Figure 4.9 illustrates the process of generating error vectors through a sequence of images.

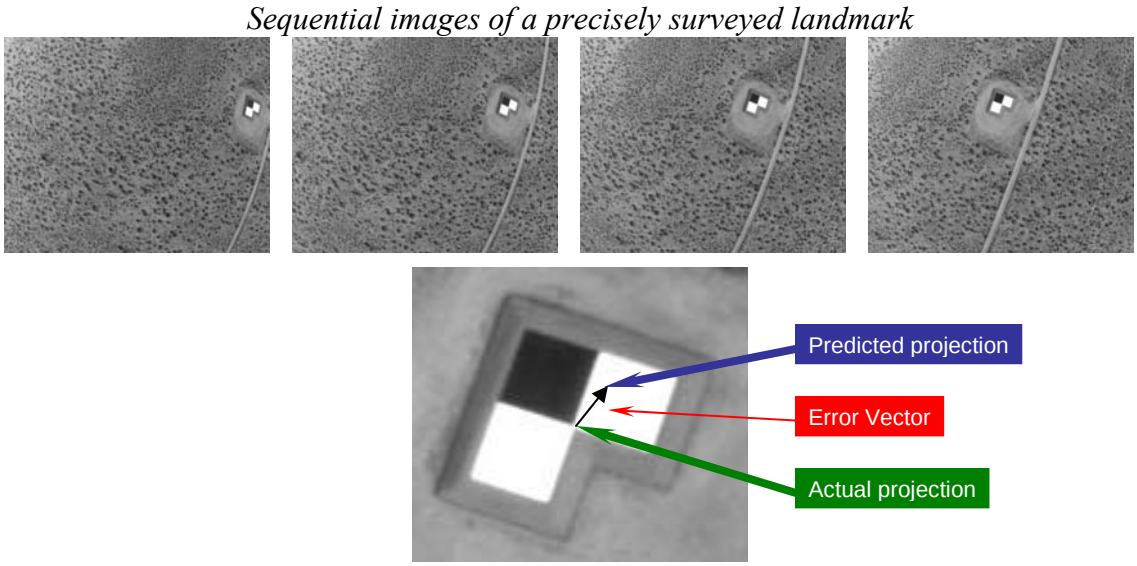

Figure 4.9: Tracking surveyed landmark to generate calibration error.

Several landmarks in the Simulated Combat Airfield and Tactical Target (SCATT) Range of the PRR were used. Sets of error vectors were formed, spanning all four quadrants of the camera frame and radial distances. Because tracking is accomplished manually, these errors are independent of the feature generation program. The resultant error vectors were fit to a vector field to produce a model of the errors induced by the optics, alignment (discussed later), and the pinhole camera model calibration (the latter being the largest contributor after calibration is complete). Figure 4.9 contains the results from the calibration of one of the PL-A741 cameras. Both cameras demonstrated statistically similar errors, and were modeled the same.

For use with a Kalman filter, the errors should be well-modeled by independent, Gaussian random variables, or have a small magnitude. The error vectors are definitely not independent or normally distributed. However, the errors are small, with a maximum

of 2.0 pixels and standard deviation of 1.1 pixels. Figure 4.11 illustrates the absolute magnitude of the errors (from the same camera). Due to the low magnitudes and quantization effects of digital images, it was determined that, for the purposes of the Kalman filter, the errors could be well modeled by a zero-mean Gaussian random variable with a standard deviation of approximately 2 pixels. Considering the computational cost and many other, much more significant sources of error (such as feature generation), storing this error map to subtract these errors was not judged worthwhile.

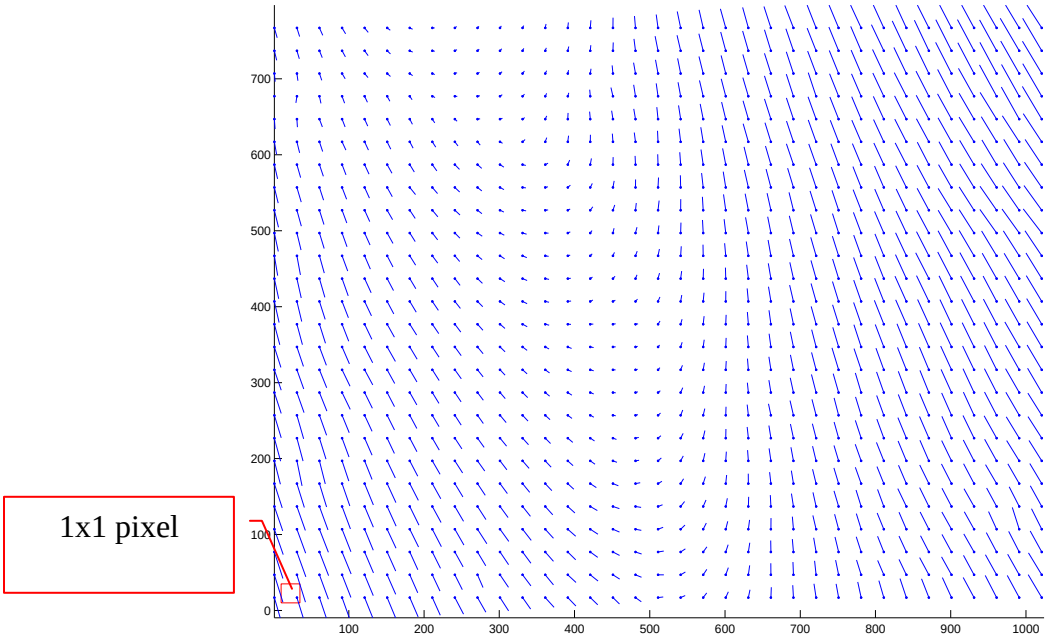

Figure 4.10: Errors induced by optics and pinhole camera calibration.

The dots in Figure 4.10 represent the base of the error vectors. The small square (bottom left) represents 1x1 pixel error magnitude. Note this is a different scale than is

used for pixel location. The error vector magnitudes are exaggerated for viewing. The X and Y axis represent pixel location in the image.

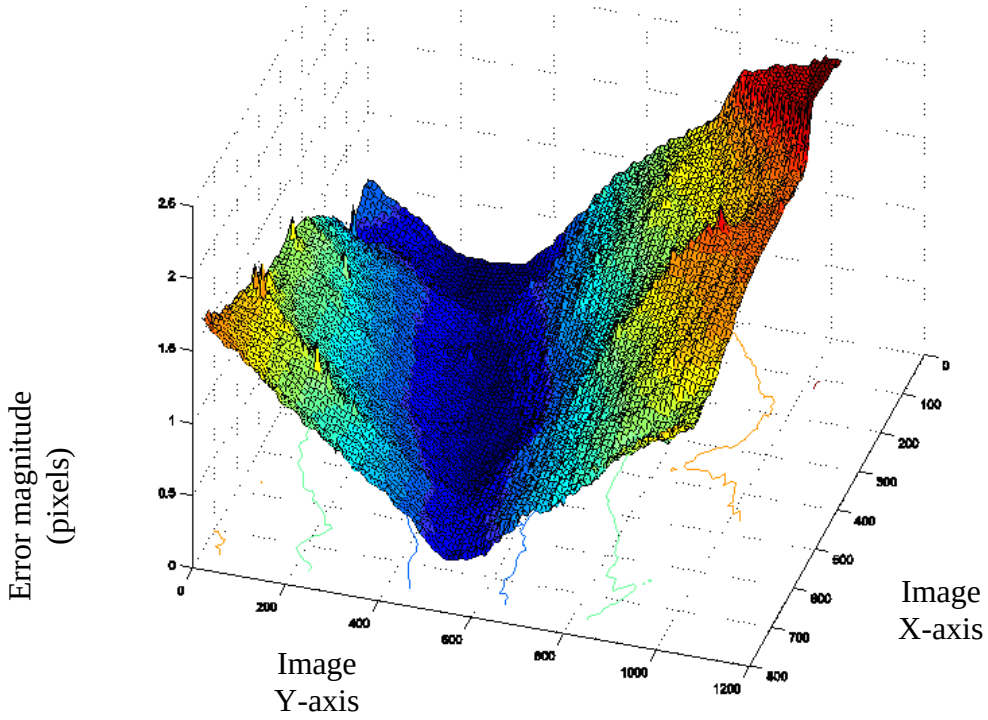

Figure 4.11: Errors induced by optics and pinhole camera calibration (magnitude).

4.3.3 Sensor Boresight. The term boresight refers to measuring the angles between the camera, IMU, and TSPI frames. These angles are then used to create the DCMs required to transform vectors (between frames).

The resolution of the cameras called for a boresight accurate to 0.1 degree in order for errors to be on the order of 1 pixel. The boresight was accomplished before

flight using precision survey equipment and personnel furnished by the AFFTC, meeting this requirement. The boresight was validated through flight test by over-flying surveyed landmarks, as described in the previous section. The boresight was used in the determination of the pinhole camera calibration error model. Errors in the boresight may have contributed to the error attributed to the pinhole calibration, but since the magnitudes were so small (1 pixel standard deviation) the boresight was considered very accurate.

The same process was repeated for the camera looking forward/aft. Iteration was used to resolve the camera pivot angle (look angle), as it was the only angle not surveyed. Again, cumulative errors were on the order of 1 pixel (standard deviation), and the look angle boresight was considered very accurate. By virtue of small cumulative errors expressed in the image projections, all boresight angles and values provided by the ground test were considered very accurate.

4.3.4 Feature Generation. The SIFT© transform was used as the primary feature generation, and ultimately matching, algorithm. It has proven quite robust in similar work by Veth [26]. The environment experienced during the test proved to be challenging however. All flight test data were collected while flying over California high desert and urban settings. The combination of the scenery's makeup, lighting, and contrast led to an order of magnitude more features per image than Veth encountered when testing over Ohio towns and farmland (4,000-9,000 features per image). This number of features was too much to process, so they were filtered by scale (a measure of size provided by SIFT©). The lowest scale features were thrown out to yield 400-600

features per image. SIFT© proved very capable of producing ample features in even the most sparse sections of the Mojave Desert. A lack of features was never a limiting factor on performance. This wasn't the case over water or dry lakebeds, where contrast was so low, that no features were produced.

4.3.5 Feature Persistence. The SLAAMR algorithm provides a navigation solution, but requires landmarks to be tracked within a sequence of consecutive images. The SLAAMR algorithm itself does not do any image processing, but relies on the SIFT© algorithm do feature generation and ultimately, matching. As discussed in Chapter 3, a feature exists in an image and corresponds to a ground stabilized point. The SIFT© algorithm generates features in images, each with a unique descriptor that correlates strongly to the same feature in a successive image.

Initial analysis of frame-to-frame feature correlation (for the flight test data) showed poor feature persistence between frames. Figure 4.12 shows two sequential frames from a test flight over urban terrain.

The lightly colored dots represent features generated by SIFT©. The crosses represent features selected for matching (reduced set for readability), and the lines between the crosses represent the best match in the subsequent frame, according to Equation (3.13). In the ideal case, the lines would all be parallel and of near equal length. As can be seen in the figure, there are some false positive matches, and these are associated with low correlation coefficients. Through careful feature-by-feature examination, it was determined that the vast majority of false positive matches had

correlation coefficients less than 0.9. This is not an absolute rule by any means, but provides a sound decision model.

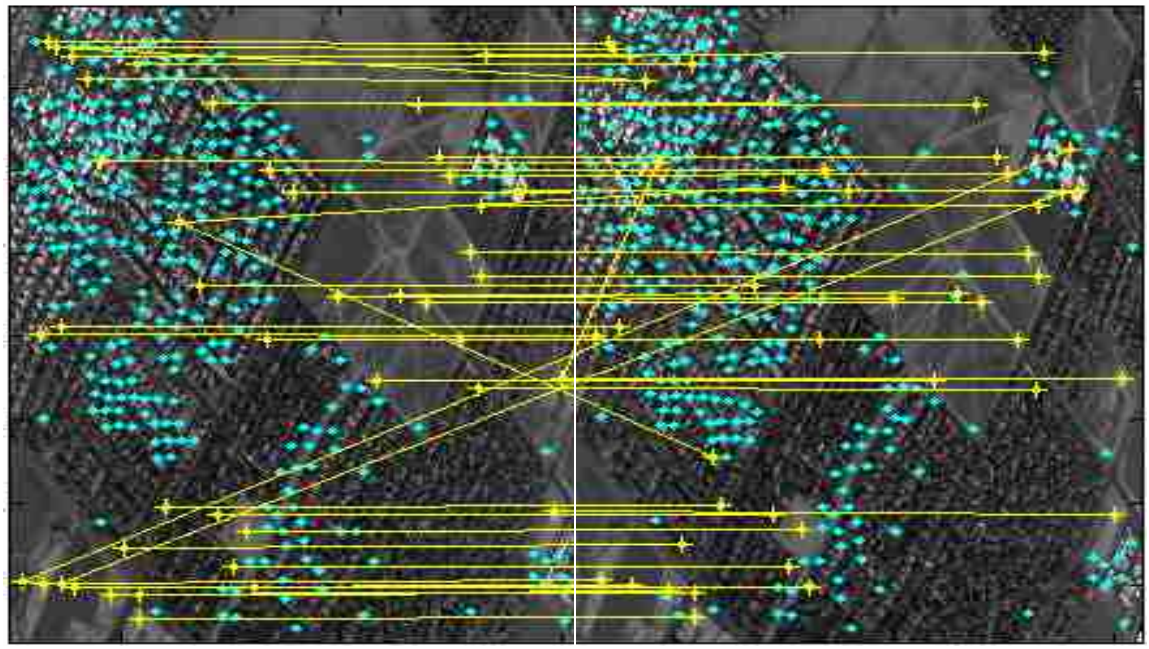

Figure 4.12: Sequential, side-by-side urban images marked with features and correlated matches.

Figure 4.13 illustrates the consecutive frame-to-frame feature correlation coefficients for 15-second samples of images, plotted as histograms over time. The correlation coefficients for all features identified in the image are plotted in 20 adjacent bins on the horizontal axes against the frame number. The correlation coefficients represent the values of the '*best'* matches in the subsequent frame. The vertical axis indicates the frequency of occurrence of an individual correlation bin. The frequency axis is the normalized fraction of all feature correlations that fell into the respective bin.

The frame number axis is meant to show trends as a function of time (frame-to-frame behavior).

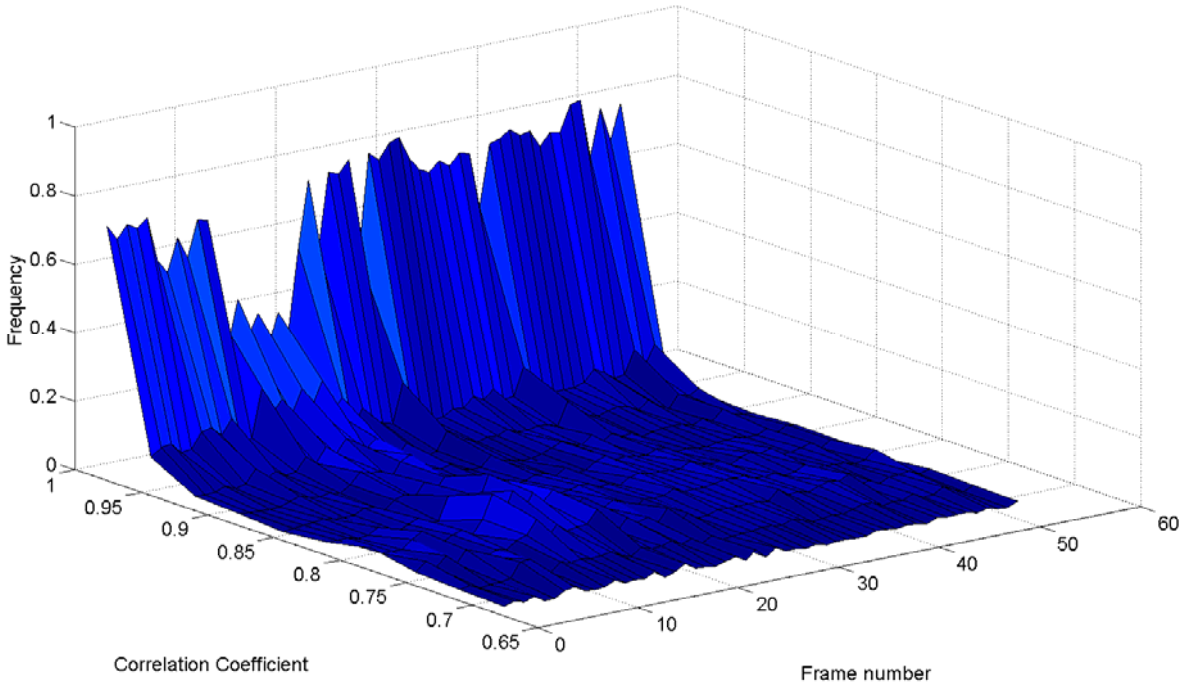

Figure 4.13: Frame-to-frame Correlation Coefficients.

A correlation value of 0.90 or greater appears to be a good choice for the "Positive Match" threshold. Repeating the process shown in Figure 4.11, and eliminating correlations below 0.9, supports the choice of 0.9 as the threshold. This choice minimizes (experimentally) both alpha and beta errors (false negative and positive matches). The SLAAMR algorithm does not simply search for the best match in the next frame. It employs a spatially constrained search using the stochastic projection of the feature's estimated position (as described in Section 3.2.2). The resultant search volume

yields a small subset of features for match consideration immediately surrounding the estimated projection. If a *true* match is available, it will be included in this subset of features. The nature of the correlations was such that *true* positive matches have coefficients ranging from 0.9-1.0 and *false* positive matches ranged between 0.7-0.9. The spatial search constraint did nothing to increase *true* positive matches, but minimized *false* positive matches when the *next best* match had a coefficient near the threshold of 0.9 (*next best* matches statistically did not fall inside the search volume).

The problem introduced by these test results is that, as a best case, only 80-90 percent of the features have matches in the subsequent frame. This suggests a possible 10-20 percent loss in features tracks for every frame. Some of the lost matches are due to a small number of features falling out of the field of view, but as seen in Figure 4.12, this is not the driving factor (this is a slow speed, therefore, small translation condition). If this loss is random and cumulative, the tracker would expect to drop all tracks in short order. The behavior over many frames must be evaluated.

Figure 4.14 compares histograms of the correlation coefficients with increasing frame spacing. To provide navigation aiding, a feature must be tracked consistently over many frames of video. The histograms show that the number of positive matches (correlation coefficient >0.90) decreases and the number of rejected matches or lost tracks (correlation coefficient <0.90) increases as the frame spacing increases.

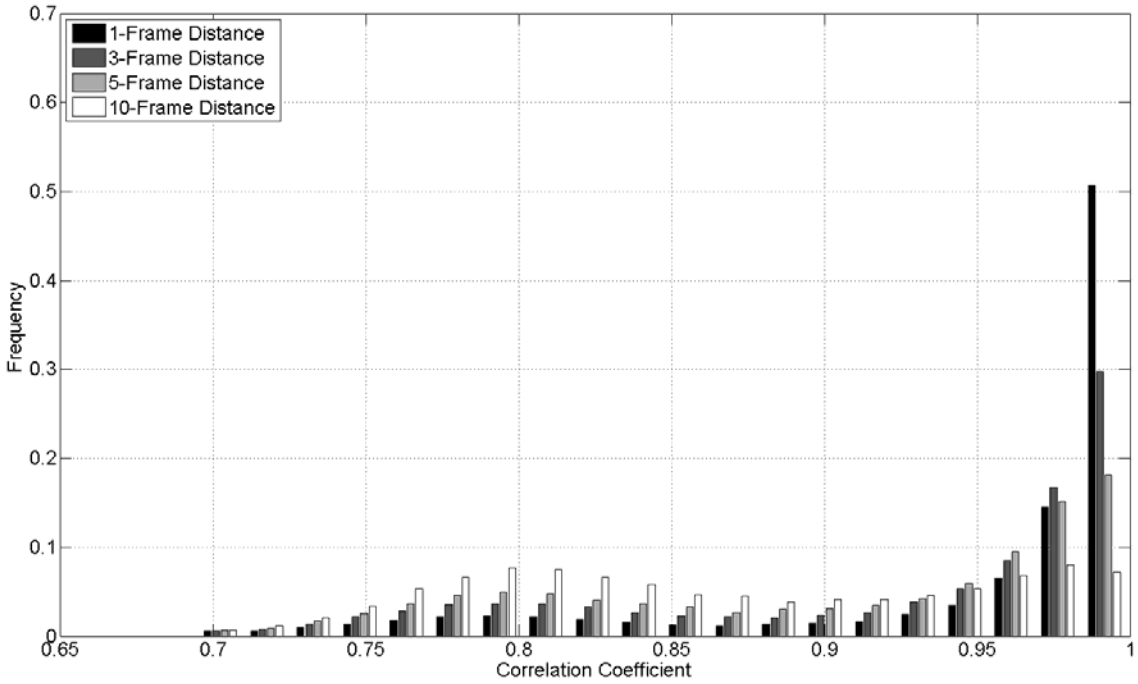

Figure 4.14: Correlation coefficients with increasing frame interval spacing.

As frame spacing increases, the mass shifts from right of the decision threshold, to left. The effective loss is not cumulative, but this does illustrate that feature persistence is a problem. By the tenth frame, most of the features do not have a match, by the stated criteria. In either case, either the feature generation software and/or the technique used to match features demonstrate poor performance with regard to frame-to-frame persistence. Two mitigating techniques are used to isolate the SLAAMR algorithm's performance from the questionable feature generation (and/or matching) persistence.

4.3.5.1 Descriptor Handoff. The single frame interval persistence, though not perfect, greatly outperforms the 2 and 3 frame intervals. For this reason it is beneficial to exploit the single frame persistence with every match. This is done by 'handing off' the feature descriptor every time a match is made. The algorithm does not compare features to the original descriptor found when the track started, but to the descriptor from the most recent frame. The descriptor to match is handed off to system memory along with the measured projection and estimated position. This technique could lead to descriptor wander (and therefore track wander), but the spatially constrained search volume (provided by the predicted landmark projection) minimizes this effect.

4.3.5.2 Non-Causal Look-Ahead. The frame-to-frame persistence problem is not driven by a random frame-to-frame attrition. Thus it can be asserted that some features naturally persist better than others. No online method was determined (despite plenty of efforts) to determine which features fell into this favorable category. The non-causal look-ahead method involves taking a feature in one frame, and searching future frames to ensure that it does, in fact, persist. In the absence of a time machine, this technique can be accomplished off-line (post-processing) or with a fixed interval smoother [14][15][16]. Essentially, this involves running the aided algorithm at a fixed interval (that required for look-ahead), and inertially propagating (from the most recent aided solution) into real time. Experimentally, looking ahead 2 frames and applying the same match decision threshold of 0.9 (correlation coefficient value), proved to mitigate most of the persistence problems encountered. Two frames at 4 frames a second, yields a

½ second delay due to the fixed interval smoother; an acceptable interval for a real-time navigation routine in this application.

A fixed interval smoother is not used in the experimental results herein. The look-ahead is merely used as a technique to isolate the SLAAMR algorithm from poorly performing image processing techniques, which are not the focus of this study.

4.4 Flight Plan

The flying portion of the test had one goal: to collect data. The data collected was then used to evaluate the SLAAMR algorithm (and the factors influencing performance) experimentally. The flight plan was simple: fly a constant velocity and altitude over a desired area. The majority of the data evaluated herein was collected by over-flying the cities of Palmdale and Lancaster, CA, at 3,000-4,000 ft above ground level (AGL). The airspeed was modulated to achieve 190 knots of ground speed, and monitored with a portable GPS unit. This nominally gave a *V over h* ratio of 0.08-0.12 (sec^{-1}). The terrain was generally flat and populated with urban features. Figure 4.14 illustrates the general flight path, which lasted approximately 40-50 minutes.

For every flight, two cameras were carried, set to two different angles (1 directly down, and 1 either 45 degrees forward or aft). This allowed a side-by-side comparison of the effects of camera look angle on aiding strength. In order to evaluate the effects of tracking differing numbers of landmarks, either surveyed or landmarks of opportunity, the data from each flight was merely processed differently. No additional flights were required. In order to evaluate the effects of *V over h*, different altitudes were flown. The airspace did not allow this to be accomplished over the city, and was therefore

accomplished over the uninhabited desert. This did not allow a simple, single variable study of *V over h* (urban features and desert features are different in nature), but a general trend was evident. The flight test data were partitioned into 10 independent 5-minute segments. Theses 10 segments were used to build Monte Carlo plots and capture the statistical nature of the navigation errors.

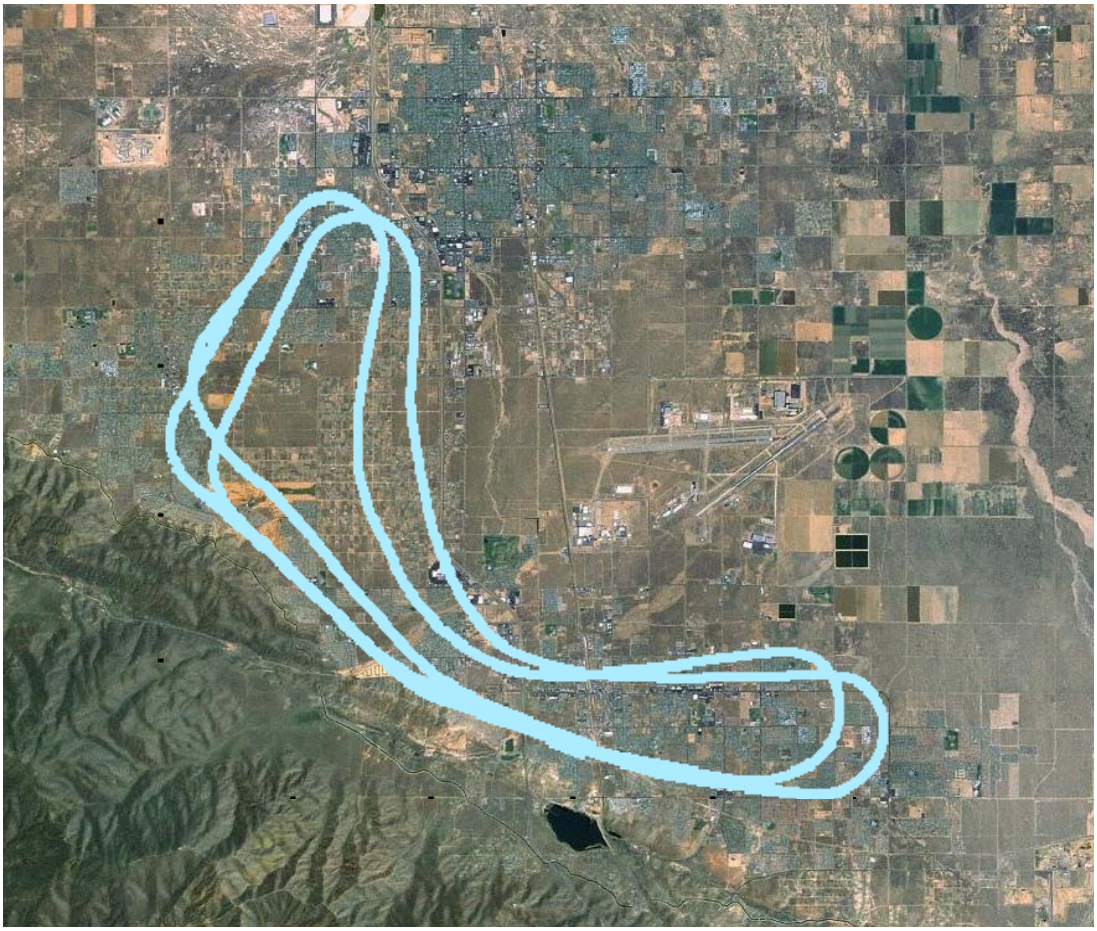

Figure 4.15: Flight path flown over Palmdale and Lancaster, CA.

The collective, 2 sigma error (2 standard deviations) for each measure are plotted against time in the following sections. In other cases, 30-minute continuous data

runs are examined. This provides a long term evaluation of the algorithm's performance, but is a single instance, and no statistical significance can be gathered.

Other profiles were flown to collect data during operationally representative maneuvers. These profiles do not contribute as much to the pure characterization of the algorithm, but speak to the operational viability of the system. These profiles include flying lower over the desert to achieve a higher *V over h*, and straight-in landings.

4.5 Results

This section presents and evaluates the results gathered from flight test. When appropriate, simulated results will accompany flight test results, and comparisons will be made. Flight test results will address the propositions and design decisions made in Chapter 3. Test data will be used to:

- Validate the assertion that inserting surveyed landmarks into an extended Kalman filter (that augments the navigation state vector to estimate landmark position) is the practical equivalent to assuming position is known.
- Prove that only two tracks (of known or surveyed landmarks) are required for full observability and strong aiding action.
- Validate the assertion that the EKF will naturally self-survey landmarks.
- Evaluate zero a priori knowledge landmark tracking for navigation.
- Demonstrate the benefit and necessity of augmenting image-derived bearings measurements with additional information (altitude and heading as well as digital terrain data to estimate initial landmark position).

- Demonstrate the benefit of tracking many landmarks when positions are unknown.
- Demonstrate the effect of look angle on aiding strength, and validate the assertion that looking forward provides the poorest performance (down or backwards is optimal).
- Demonstrate the effect of *V over h* on aiding action.

This chapter will also demonstrate the fluid nature of the SLAAMR algorithmic design, allowing seamless transition between tracking modes (LOO and surveyed tracks). This will be shown through demonstrating navigation with a partial set of surveyed landmarks. The results from an operationally representative landing profile will be discussed. Lastly, the viability of a passive, optical-inertial navigation system will be demonstrated with a long duration flight that performs comparably to a conventional navigation grade INS.

The primary measure of performance considered is the overall drift of the navigation solution. The drift will be quantified by examining the circular and spherical position and velocity errors over time, as well as individual attitude state errors. There are many measures to evaluate these errors, but the mean and standard deviations will emphasized herein. Specifically, the absolute mean summed with two times the standard deviation (mean plus 2 sigma) is used as an approximate 95% confidence bound for the errors. The errors, specifically the circular/spherical position and velocity error, were non-zero mean and plagued with many systematic characteristics (created by image processing and tracking performance). They did not conform to any conventional

random variable. The mean plus 2 sigma measure provides a consistent confidence interval for the data collected, where other measures (such as root-mean-square) did not perform as well (in this experiment). When applicable, single influencing factors will be varied to show the impact on performance.

The overall goal of this report is to characterize the true aiding potential that tracking landmarks can give to an airborne INS. Additionally, it aims to expose the challenges of such a problem.

4.5.1 Effect of Surveyed Landmarks. The assertion was made that tracking well-surveyed landmarks yields performance near that of tracking perfectly known landmarks. This will be shown to be true despite not having a full-rank observability grammian matrix. In the perfectly known case, the attitude angular errors are on the order of the camera/feature generator angular resolution. The position and velocity errors are bounded and small. Figures 4.16 through 4.18 display the results of three sets of Monte Carlo runs, where 2, 3, and 10 surveyed landmarks are tracked at any given time. The three runs span each look angle (forward, downward and rearward).

Each Monte Carlo run consisted of 8-10, 5-minute data segments. The aircraft was flown at approximately 1000-1300 meters, at approximately 100 m/s, over flat urban terrain as described in Section 4.4. The mean plus 2 sigma (2 standard deviations) spherical position, velocity, and attitude errors are displayed for each condition.

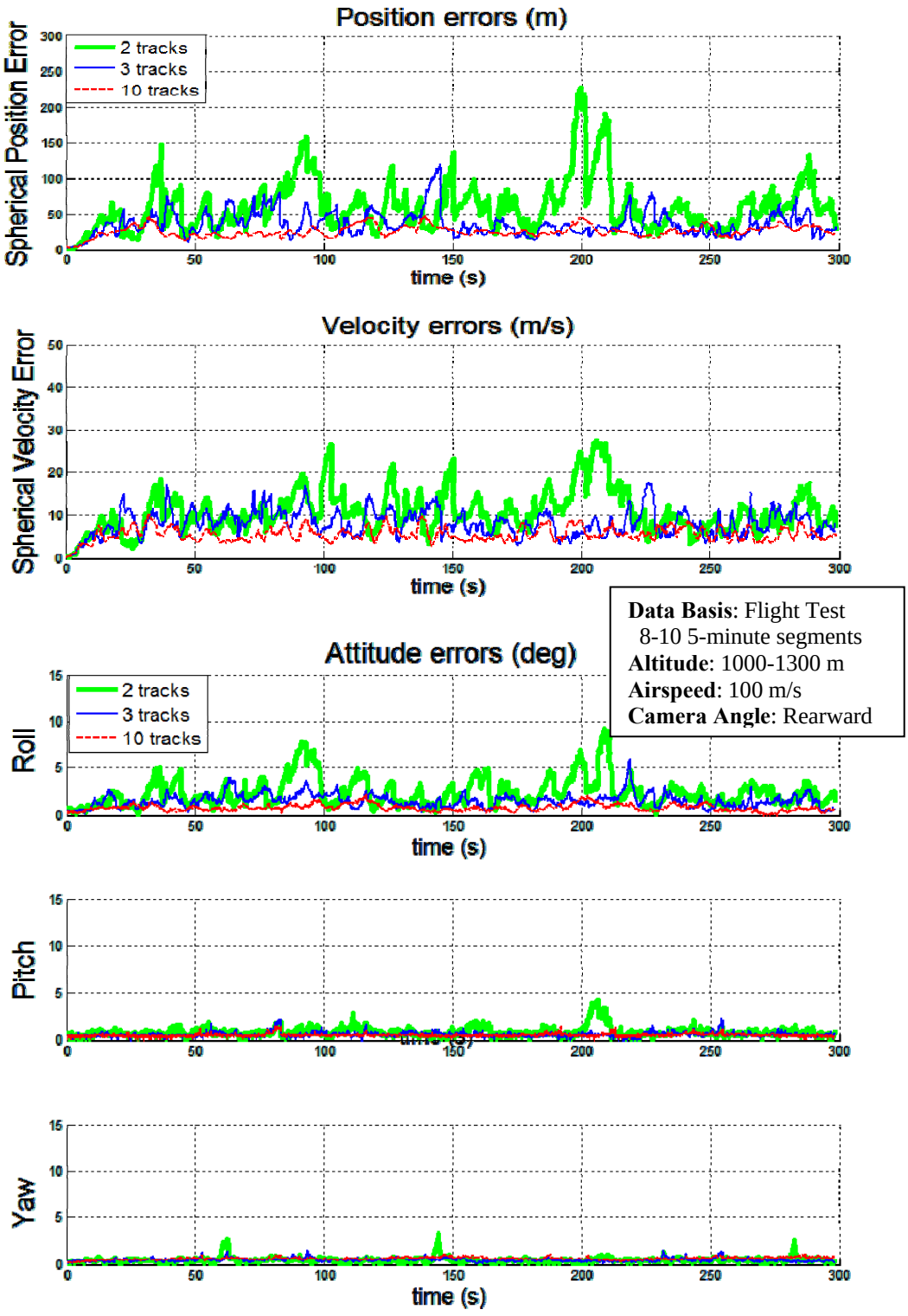

Figure 4.16: Tracking surveyed landmarks with REARWARD looking camera.

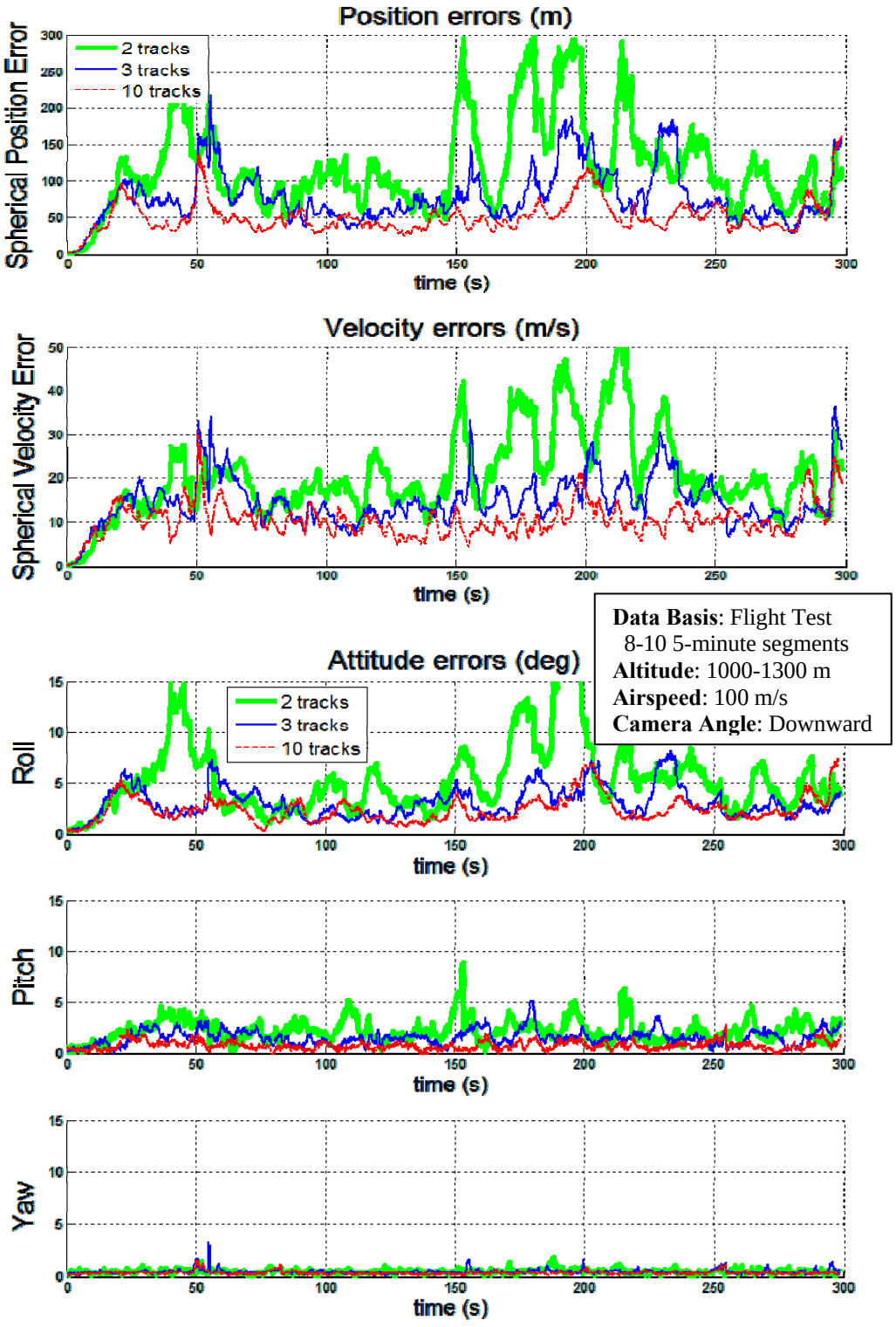

Figure 4.17: Tracking surveyed landmarks with DOWNWARD looking camera.

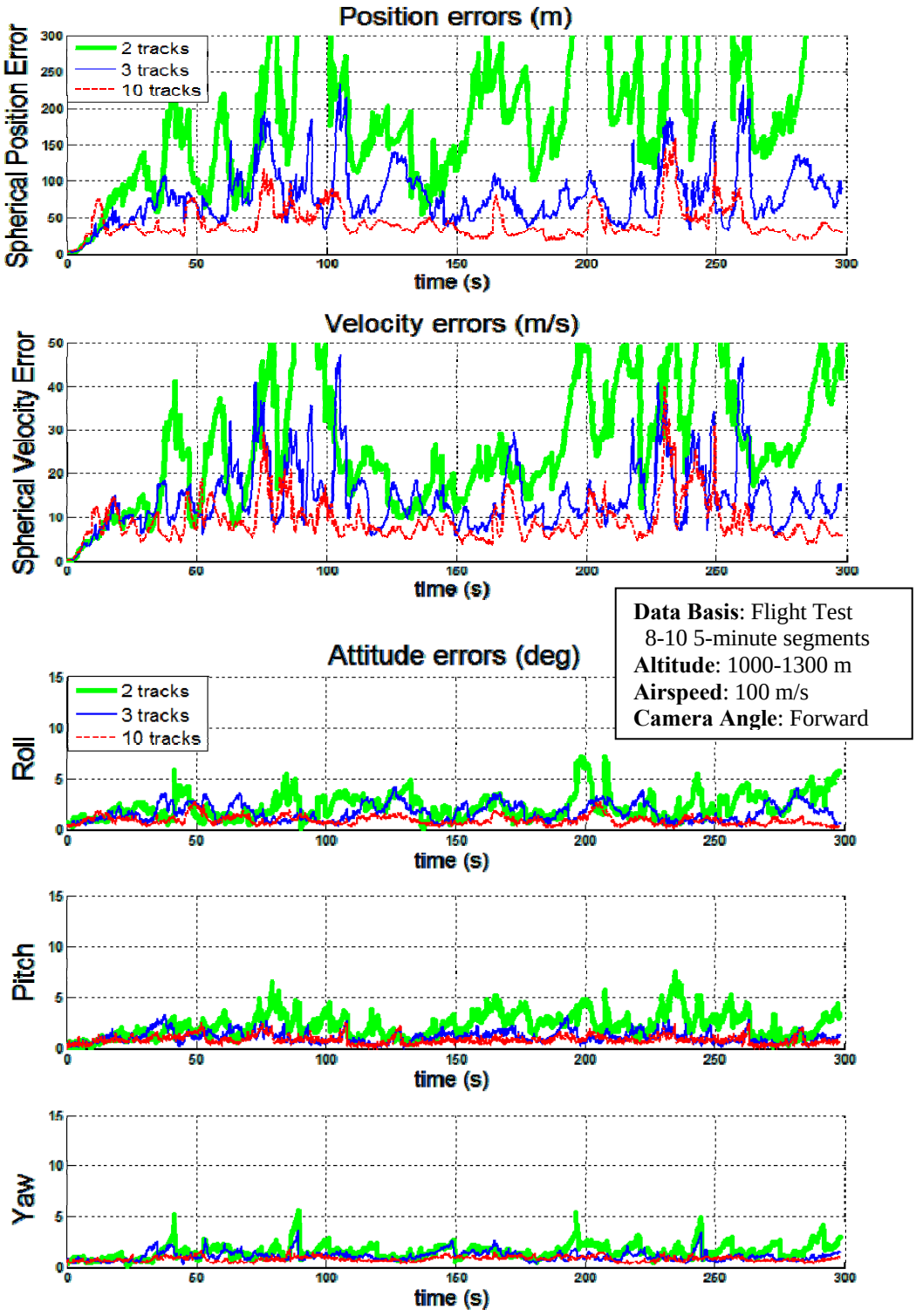

Figure 4.18: Tracking surveyed landmarks with FORWARD looking camera.

In all cases, the errors were bounded and relatively small, but there was a distinct performance difference. The rearward looking camera outperformed the downward and forward in all cases by a significant margin. The rearward looking camera errors are approximately half of those for the downward and forward cameras by all measures. As described in Chapter 3, the rearward camera provided more favorable feature flow, guaranteeing better GDOP and condition number. When 3 or more surveyed landmarks are tracked, performance is greatly improved. All conditions showed stable errors (with respect to time). The errors fluctuated, but did not tend to drift and grow. Tracking surveyed landmarks provide a stable navigation solution, implying *virtually* full observability, matching drift behavior predictions for tracking known landmarks.

The rearward looking camera condition performed the best overall. As can be seen in Figures 4.16 through 4.18, there is a slight performance advantage to tracking 3 versus 2 surveyed landmarks. There is only a marginal advantage to tracking 10 versus 3. The downward and forward camera angle conditions showed worse performance than the aft, but that is due to greater GDOP. Tracking 2-3 surveyed landmarks also demonstrated poorer overall performance than tracking 10. Due to so few landmark tracks, the GDOP fluctuated as the tracking geometry changed. This caused momentary error excursions, which recovered as the GDOP later decreased. Tracking 10 landmarks provided better average GDOP over time, and thus fewer excursions. These GDOP artifacts would be present if tracking perfectly known landmarks (assuming a real camera is used).

As predicted in the perfectly known landmark position case, flight test attitude errors are generally on the order of the optical/feature generation resolution. The collective optic and feature generation noise (corrupting bearings measurements) was modeled to be 2 pixels standard deviation, which is equivalent to approximately 0.2 degrees. When tracking 10 landmarks looking aft, the 2 sigma attitude errors are on average 0.4-0.5 degrees (Figure 4.16), making the standard deviation 0.2-0.25 degrees. Tracking fewer landmarks caused excursions away from this lower limit of 0.2 degrees, but generally returned. Attitude error magnitude depends on the dimension examined. The heading errors are on the order of the camera resolution (0.2 degrees standard deviation or sigma). Pitch errors are complicated by the fact that the aircraft is traveling forward; feature translation due to forward velocity is ambiguous with pitch rate. For the test conditions, the feature angular translation at nadir due to forward velocity (frame-to-frame) is approximately 1.5 degrees. For the poorly performing downward looking camera case, 1.5 degrees is approximately the magnitude of the pitch standard error. The forward and rearward looking cameras benefit from tracking landmarks near the horizon. On the horizon, all feature translation is due to rotation, and helps to resolve the pitch rate and velocity ambiguities. Roll errors are naturally higher, since none of the conditions capture a lateral horizon and the tracked landmark geometry trended along the longitudinal axis. GDOP is naturally poorer about the roll axis. These GDOP effects would be present in the perfectly known landmark case, thus the flight test data agrees with prediction. More precise optics and feature generation would naturally yield better attitude performance.

The magnitude of the navigation error is a function of GDOP and optics. For the best GDOP condition (the rearward looking case tracking 10 surveyed landmarks), the standard error (1 sigma) was approximately 12-13 meters. It was shown that an aircraft positional uncertainty (when using perfect optics) will approach that of the tracked landmark. If the landmark position is perfectly known, the lower limit is zero. Since the camera used was not perfect, the lower limit is driven by the optical resolution, as projected on the landmark. For this flight test example, the 0.2 degree standard error camera, projects a 3.5-4.5 meter standard error on the ground, 1000-1200 meters away. The camera resolution error is on the order of the position errors, but there is more to consider. The attitude uncertainty also contributes to the lower limit bound for perfectly known landmark tracks. In the known landmark position case, the attitude errors are limited by the camera angular resolution. In the best case, this adds another 0.2 degree, and 3.5-4.5 meters of standard error on the ground. Considering this case, the flight test data agrees reasonably well with the perfectly known case. Notably, flying higher with the same camera will result in larger position errors, and vice versa.

Velocity errors require a more complex examination. The velocity states are not included in the measurement equation. Velocity state aiding comes via their correlation with position errors; position is the integral of velocity. For velocity to be aided, the velocity errors must influence the measurement residuals (must be observable and have reasonable grammian matrix condition with respect to the velocity states). The residual errors are expressed in the camera frame as pixel or angular errors. Residual errors are caused by estimated landmark projection errors; a function of attitude errors, optical

resolution, aircraft and landmark position errors. If the landmark position is known, and the attitude estimate is precise, it leaves optics and aircraft position error. The position error in the residual comes from drift of the position estimate from one frame to the next. By definition, this is velocity error. A single measurement cannot discern velocity contributions below the optical measurement noise. Thus, the optical error standard deviation, projected onto the ground, sets the threshold for resolvable velocity error standard deviation multiplied by the time between frames. In the test data, the 0.2 degree standard error camera projects a 3.5-4.5 meter standard error on the ground. At four frames a second, the threshold for velocity error standard deviation is 14-18 m/s. This discussion refers to the resolution of the measurement equation only. The Kalman filter accounts for all measurements taken over time, the system dynamics, and balances it against the process driving noise. When all are considered (using the propagation and update equations from Chapter 2), and the actual flight test conditions are applied, the predicted steady state spherical velocity error is 11-12 m/s (or 22-24 m/s when best case attitude errors are considered). These predictions agree with the test data. Thus, the relatively large velocity errors seen in Figures 4.16 through 4.18 can be attributed to the camera resolution, and would be present even if landmark position was perfectly known. Needless to say, the velocity errors gathered from flight test data agree with predictions for tracking known landmarks. Notably, flying higher with the same camera will result in larger velocity errors, and vice versa. This discussion does not speak to observability, but to the resolution of the measurement equipment and the effect of aiding performance.

The goal of this section was to validate the assertion that tracking surveyed landmarks yields the practical equivalent to assuming position is known. Flight test data clearly show that this is true if the conditions are right (camera look angle and/or number of tracks being good). Since GDOP is the culprit behind the poor performance, perfect knowledge of landmark position will do no better (assuming that a real camera is used, without infinitely fine resolution and with non-zero error).

The landmarks were surveyed using truth data and DTED. The surveyed landmark position spherical errors were on the order of 3-5 meters (standard deviation). This is a relatively poor survey, but still yields quality aiding nonetheless. Flight test results imply that, though a highly precise survey could benefit, it isn't required. The spatial sample size, driven by a single pixel (roughly 0.1 degree wide, 2-4 meters projected on the ground) cannot resolve much more. Add to that, corruption in the feature generation process. This of course is dependent on altitude during flight (distance from the landmark). A 5-8 meter spherical uncertainty would be useless for an aircraft flying 5 meters off of the ground, but a 20 meter uncertainty would be overkill for the space shuttle in orbit. Additionally, it was suggested in Chapter 3 that the aircraft position uncertainty would approach that of the surveyed landmarks being tracked. The 2 sigma spherical position errors in Figure 4.16 are not surprisingly close to those of the surveyed landmarks. The 2 sigma spherical position errors are on average 20-25 meters, making the standard deviation 10-12.5 meters. In fact, when the same landmarks were surveyed with less precision, the magnitude of the nav-state errors grew accordingly, but

did not drift over time. The quality of the survey merely drives the magnitude of the navigation state errors, but any independent survey bounds the drift.

The descriptors associated with the surveyed landmarks were taken from the flight test data, and not generated by a third party. This meant that feature recognition was not complicated by viewing the landmarks from different poses as the descriptors associated with the surveyed landmarks matched at least one image perfectly. Therefore the surveyed landmarks would always be found. This isolated the algorithmic performance from imperfect feature generation, but a truly robust system would be required to find and recognize landmarks surveyed by a third party. Though a critical function of a robust system, it is independent of the mathematical theory discussed herein. Once again, for the purpose of this discussion, a perfect feature generation algorithm is assumed to exist to allow the specific study of the SLAAMR algorithm.

4.5.2 Full Observability with Only Two Tracks. Figures 4.16 through 4.18 demonstrate that *virtually* full observability can, in fact, be achieved with only two landmark tracks. This is implied by the fact that in all cases (where only 2 surveyed landmarks are tracked), the position errors are bounded. If full observability were not achieved, the position errors would be expected to drift. Granted, these data do not represent the perfectly known landmark position case, but that makes the assertion even stronger. If full or *virtually full* observability can be achieved by tracking 2 surveyed landmarks (using an augmented state vector), then the extension to the perfectly known case can be confidently made. Having full observability does not guarantee good aiding,

however. This is evident in the effect that camera look angle has on performance; GDOP must still be considered.

4.5.3 Self-Survey. It was proposed that the EKF-based SLAAMR routine naturally performs a survey of its environment as it tracks landmarks. This behavior is a key factor in the ability to aid INS. It was also proposed that the uncertainty of landmark position estimates shrink, approaching that of the aircraft's own positional uncertainty. The SLAAMR algorithm's ability to self-survey is demonstrated by allowing it to track landmarks of opportunity, while being aided with TSPI. Using truth data isolates the mapping performance from an errant navigation state and poor linearization. The *Far Guess* landmark initial position estimation routine is used to locate the LOO; no a priori information about the landmarks is assumed. Figure 4.19 illustrates how the initial landmark position estimate and uncertainty volume changes as a LOO is tracked.

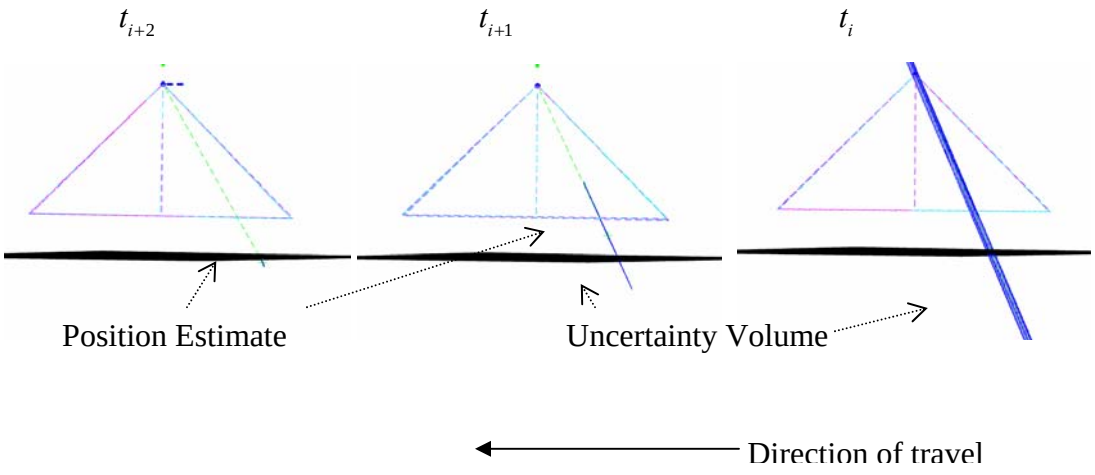

Figure 4.19: Self-survey process using the *Far Guess* technique. Three instances in time are displayed as the aircraft travels from right to left.

The triangle represents field-of-view of the camera, the dashed line points to the landmark position estimate, and the ellipsoid centered on the estimate represents the uncertainty volume of the estimate.

Figure 4.20 depicts the result of a 30-second run in which 20 LOO were tracked at any given time. A total of 60 landmarks are generated for the map.

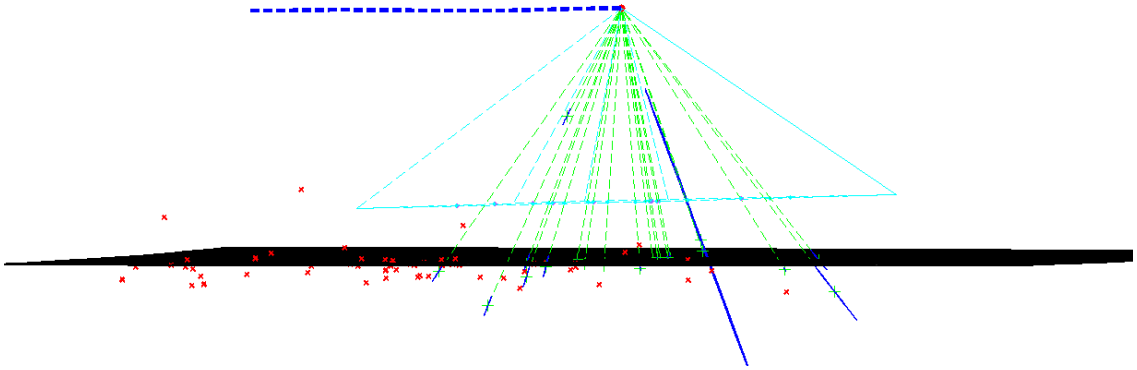

Figure 4.20: Self-survey of landmarks. The x's represent the mapped landmarks.

The 3-dimensional position of the landmark estimates are plotted against a DTED terrain map (DTED Level 1, approximately 90 meter horizontal resolution). The DTED was not used to locate the LOO, but provides a visual *truth* comparison for the self-survey. If the DTED and survey were both perfect, they would overlap. The DTED is not perfect, but is accurate enough to illustrate the point. To gain more insight, the difference between the estimated landmark elevations and DTED elevations are displayed in Figure 4.21. The errors have been sorted in order of magnitude to capture the general distribution (akin to a probability distribution function).

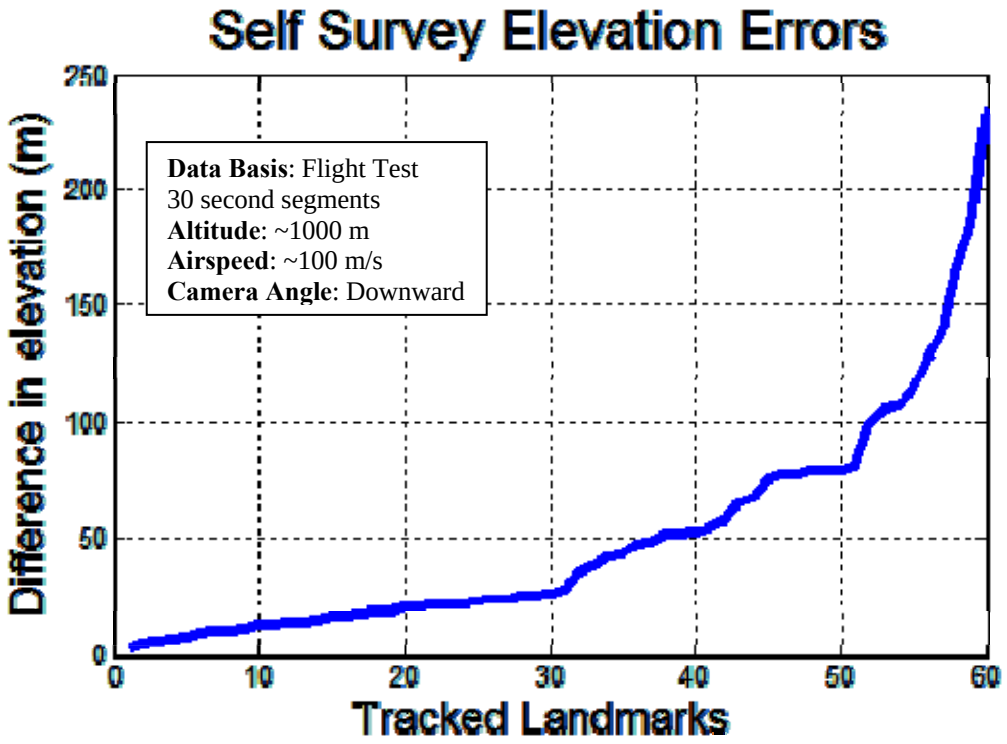

Figure 4.21: Elevation error for a self-survey of landmarks. Landmark elevation errors are sorted in order of magnitude for illustration purposes.

Because the terrain was flat and the camera used was looking down, the elevation errors are a natural measure of performance. The aircraft was flying between 1000-1300 meters above the ground. The majority of the errors (87%) fall below the 100 meter line (10% of the aircraft height). At first look, the flight test data implies that the mapping performance is very poor; the error standard deviation is approximately 50 meters. During the test, the aircraft was flown roughly 1000 m above the ground, at approximately 100 m/s. Images were taken looking down at a frame rate of 4 images-

per-second, and the combined camera/feature generator resolution was assessed to be 2 pixels standard deviation (each pixel being 0.1 degrees across). Due to the nature of the Kalman filter, the progression of landmark vertical uncertainty over time is also a function of frame rate and translation. Time is factored out, and it becomes a function of landmark apparent translation per frame and the resolution of the camera. The time dimension is replaced by total angular translation (or tracking angle). This is intuitive because as the tracking angle increases, the unobservable dimension (along the camera z-axis at the first image), becomes more resolvable. Figure 4.22 illustrates the dependence on camera resolution and tracking angle for the flight test condition under which landmarks nominally translated 1.5 degrees per measurement. The feature translation rate is assumed constant and the cumulative tracking angle builds as more images are taken. For example, a cumulative tracking angle of 15 degrees is achieved after 10 images are taken.

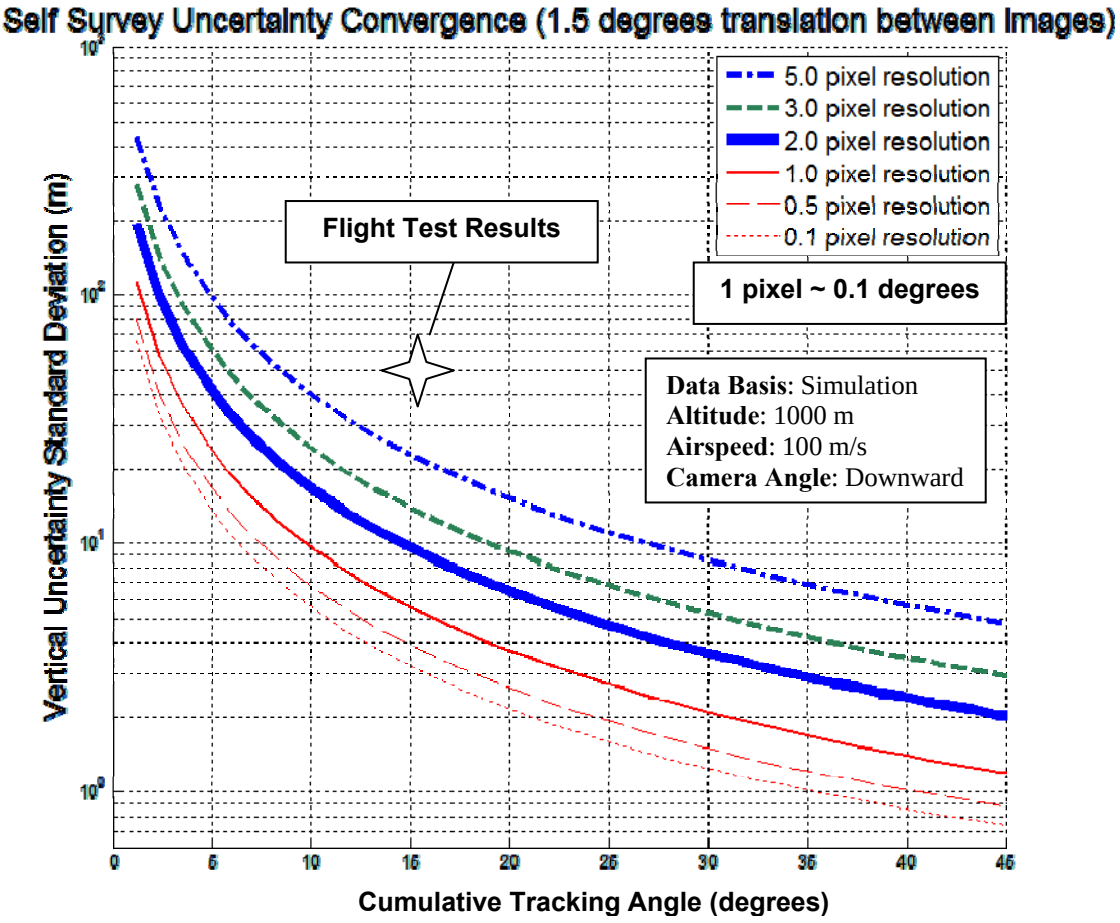

Figure 4.22: Effect of camera resolution and tracking angle on self-survey performance.

For a landmark to be included in the map (shown in Figures 4.20 and 4.21), it had to have been tracked for at least 10 consecutive frames. Any fewer and it was considered a poor track and thrown out. This meant that every landmark had been tracked for a minimum of 15 degrees (cumulative tracking angle). Some landmarks were tracked much longer, but due to the previously discussed feature persistence problems, landmarks averaged 10-15 tracks. Given that the feature was tracked from nadir on (as was generally the case for the flight test data), this would translate into a 10 meter standard

deviation; see Figure 4.22. The flight test data revealed much worse performance than expected (~5 times). This was attributed to the imperfect feature generation which noticeably misplaced features (in consecutive frames) by more than 2 pixels. The flight test data failed to support theory, but performance is limited by image processing and optics. Given a perfect feature tracker that allows a landmark to be tracked for 45 degrees without contributing to measurement error, the same optics used in flight test (~2 pixel standard deviation error) would yield a 2 meter vertical error (standard deviation). However, given the same feature tracker used in test, a 1 pixel error camera (standard deviation) would yield a 4-6 meter error in 15-20 degrees. Lastly, a 0.1 pixel error camera would yield a 2-3 meter error. The resolution of the camera, as discussed herein, refers to the angular pointing precision. This would allow for an articulated system, like a targeting pod, which would allow for fine resolution.

Frame rate effects performance as much as resolution and tracking angle. A faster frame rate allows more images in the same time or tracking angle. This translates into a smaller apparent angular translation from frame to frame. Because the measure is images-per-degree-of-tracking-angle, this can also be achieved with a lower *V over h* ratio. Figure 4.23 illustrates the effect of an increasing frame rate (per degree of cumulative tracking angle) for the flight test condition, but using a 1 pixel error camera (0.1 degree standard deviation).

A frame rate of 10 images-per-second yields a nominal translation of 0.5 degrees-per-frame (at the flight test condition). According to Figure 4.23, this would yield 0.7 meter error in 45 degrees, or 2.5-3.5 meter error in 15-20 degrees. The same effect could

be achieved via the *V over h* ratio, by increasing altitude 2.5-3 times, decreasing airspeed 2.5-3 times or any combination thereof.

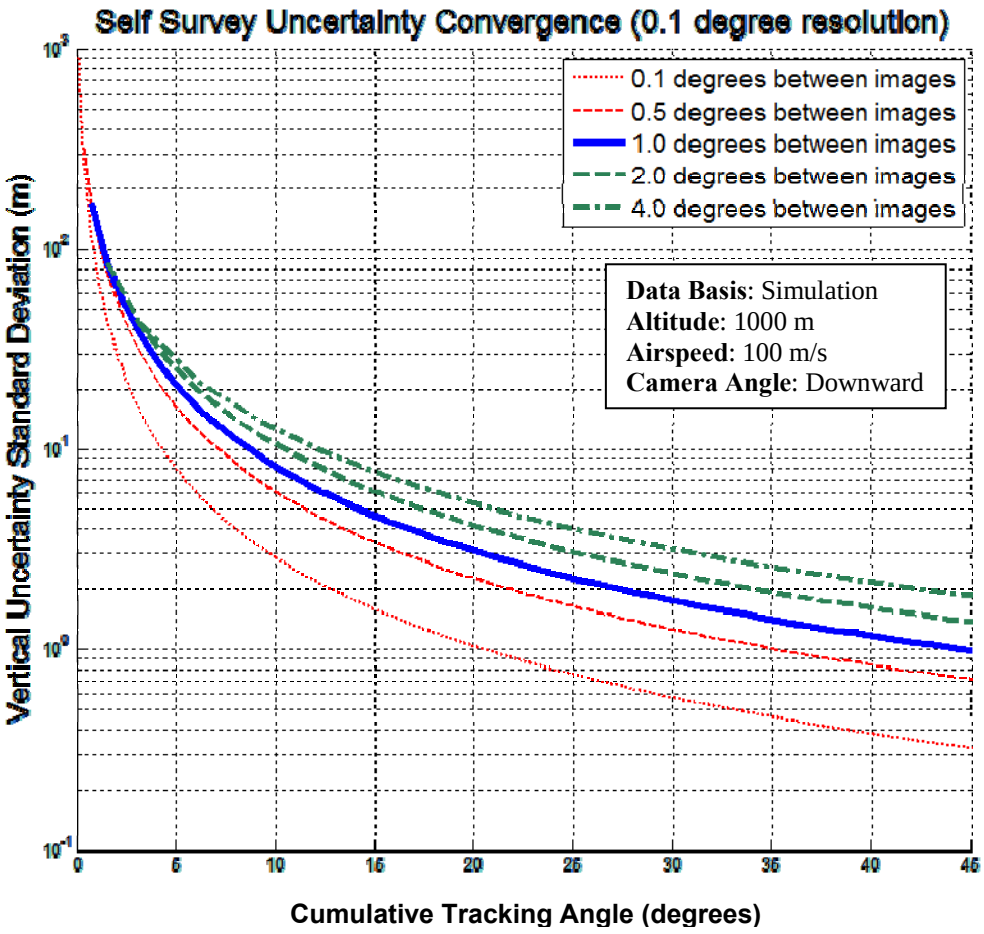

Figure 4.23: Effect of frame rate per degree of tracking angle on self-survey performance.

Figures 4.22 and 4.23 were generated through simulation, by propagating the Kalman filter according to the particular scenario. Only the error state covariance matrices were propagated (assuming perfect navigation state knowledge), yielding the

theoretical uncertainty in landmark elevation estimation. These figures represent the ideal case which can never be achieved, but test results agree.

Figures 4.22 and 4.23 clearly show the advantages of more precise optics, more consistent feature tracking (thus longer track angles), and faster frame rates or lower *V over h* ratios. Flight test data agrees with these figures, validating the model, but data were not taken at lower *V over h* conditions or with different optics. The recursive process used in the SLAAMR algorithm can, in fact, be used to build a 3-dimensional map of the surroundings. However, optics, frame rate, geometry, and feature tracking must be considered and taken into account. A deficiency in one area may be corrected by another.

When using the self-surveyed map to aid navigation, the precision of the map, and speed at which precision is achieved, greatly dictate how strong the aiding action will be. The relative magnitude of the errors in the map to those in the navigation state relate to a dilution of precision. This isn't the same as GDOP per se, but a system that cannot quickly build a precise map (before tracks are lost) cannot aid itself. The system under test, combined with the profiles flown, did not accomplish this task well; thus the data runs using the *Far Guess* estimation routine did not perform well. This will be discussed in Section 4.5.4.

4.5.4 Navigation with Zero A Priori Information. It has been proposed here and in the literature [5] that a SLAM can be used to aid an INS without the benefit of a priori information about the environment. Veth showed that this can be accomplished to some precision using binocular vision to acquire initial landmark position estimates [26].

Binocular vision requires two cameras, spaced far enough apart to resolve distance precisely in the z-axis of the *c-frame*. For airborne applications, the distance between the cameras becomes quite large as the distances requiring resolution are also large. In the case of the SLAAMR algorithm, zero a priori information (ZAPI) navigation is accomplished monocularly using the *Far Guess* initial landmark position estimation routine, and no other aiding.

For SLAAMR to achieve precision ZAPI navigation, a precise map must be built and tracked before the inertial solution diverges too far from truth. Specifically, a minimum of two landmarks must be self-surveyed to a precision near that of the aircraft's own position. As it was shown in the previous section, the flight test data did not achieve a precision map for many reasons. Figure 4.24 illustrates the Monte Carlo results of using only LOO tracks and the *Far Guess* initial position estimate routine. The mean plus 2 sigma errors are compared to the free inertial (Free Run) performance for both the downward and rearward looking camera angles. Again the aircraft was flown 1000-1300 m above the ground, at approximately 100 m/s, and each Monte Carlo run consisted of 10 segments of test data.

The data are clear; ZAPI navigation with the system under test is extremely poor. By all measures, the performance is worse than the free inertial case. The navigation errors go unstable very quickly, making landmark tracking impossible, terminating the algorithm after about a minute. There is a slight advantage to looking downward over rearward in this case, but the free inertial is always better.

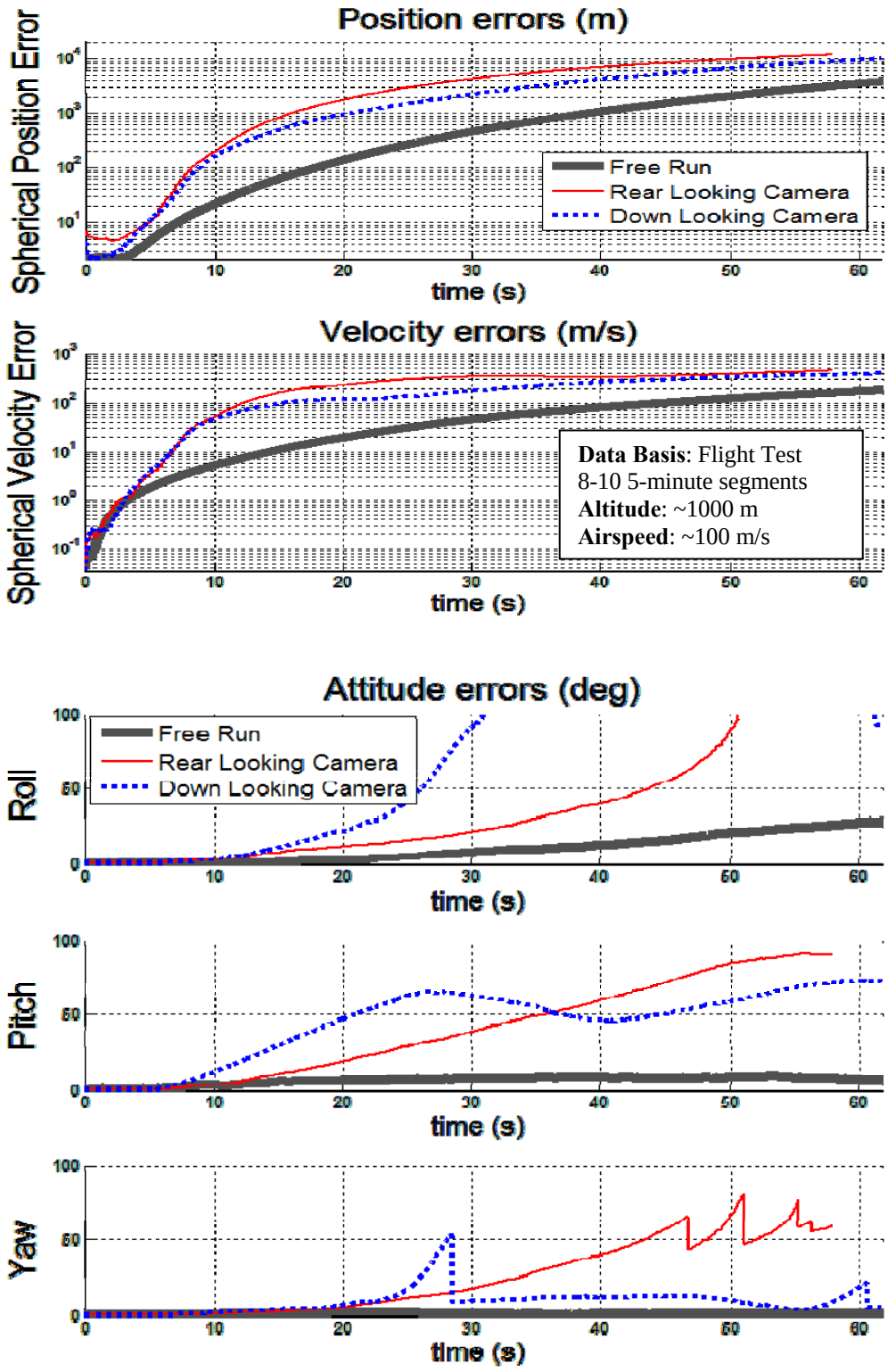

Figure 4.24: Zero A Priori Knowledge Aiding Performance.

Because the free inertial is always better, no insight could or should be gathered about why one condition outperformed the other. The estimates have diverged so far from truth that the EKF is not operating under the designed assumptions. This result, however, was expected due to the findings above. The self-surveyed maps are very poor. The landmarks are not tracked long enough. Thus their positions are not resolved finely enough to be of use. It is clear that (for the specific system under test) additional aiding is required for a viable navigation solution. Possible methods to increase the fidelity of the landmark maps were discussed in the previous section, but are not part of this test.

The flight test data showed very poor performance, but simulation suggested there was some benefit to ZAPI navigation. Figure 4.25 illustrates the simulated performance of the *Far Guess* initial position estimation routine (compared to the other routines described in Chapter 3) when evaluated for a similar, simulated flight profile. The difference in performance is likely due to the un-modeled, non-linear effects discussed throughout this chapter. A previous study achieved strong aiding action using self-surveyed maps [5]. This study employed much faster camera frame rates, lower *V over h* ratios, and was not plagued by poor feature tracking (and thus low tracking angles) due to a more reliable, but less robust feature tracker. The full-scale, representative flight environment used in this test proved too much for the flight test apparatus to overcome. ZAPI navigation cannot be proven beneficial here, but also cannot be invalidated due to success in other studies. It may be that the quality of the inertial (and other) systems used were insufficient for the geometry of the problem presented.

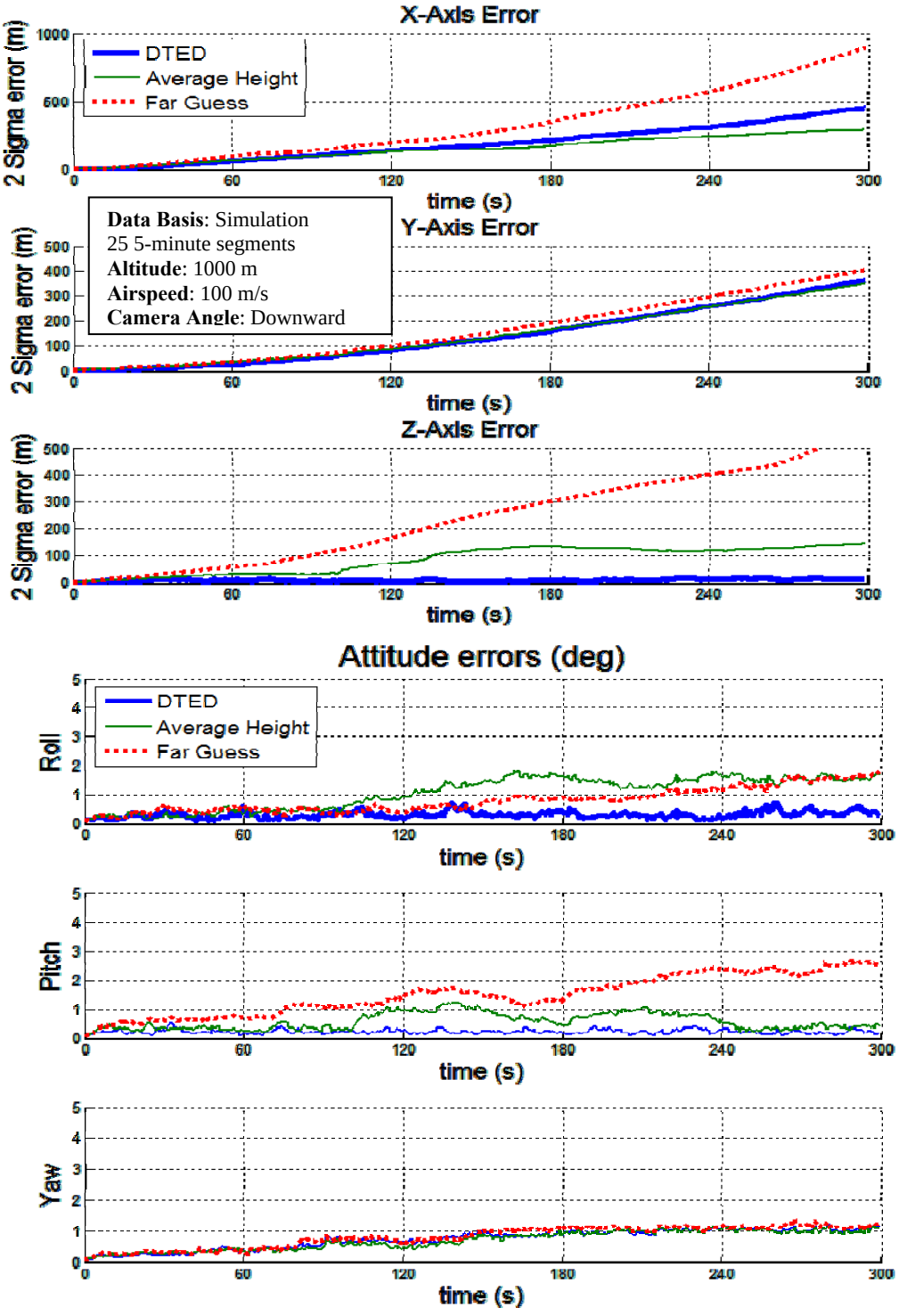

Figure 4.25: Simulated Zero A Priori Knowledge Tracking Performance.

4.5.5 Effect of Additional, Non-Image Measurements. It was shown in Chapter 3 that full observability cannot be achieved when tracking landmarks of opportunity. It was also shown in the previous section, that this form of ZAPI navigation poses a very challenging problem, producing very poor performance. However, it was also shown that measuring altitude, heading and using DTED (to estimate landmark position) significantly improves observability. Figure 4.26 illustrates the influence these measurements have on aiding. The mean plus 2 sigma errors for 4 Monte Carlo runs are displayed, each with an incremental aiding strategy. Each Monte Carlo run consists of 10 5-minute segments, using the downward looking camera, and tracking 10 landmarks of opportunity. Horizontal circular errors and attitude state errors are displayed as they provide the most insight.

The first aiding strategy considered is to use no aiding at all. The free inertial run (free run), allows the INS to propagate without the benefit of any aiding, (i.e., no landmark tracks). The following aiding strategies build up in a natural progression. The second strategy tracks 10 LOO, using DTED to estimate their initial positions. DTED could be implemented in the onboard computer without additional sensors. The third strategy builds upon the second, adding altimeter measurements. The fourth builds upon the third, adding a compass or heading measurement. The third and fourth require separate sensor hardware.

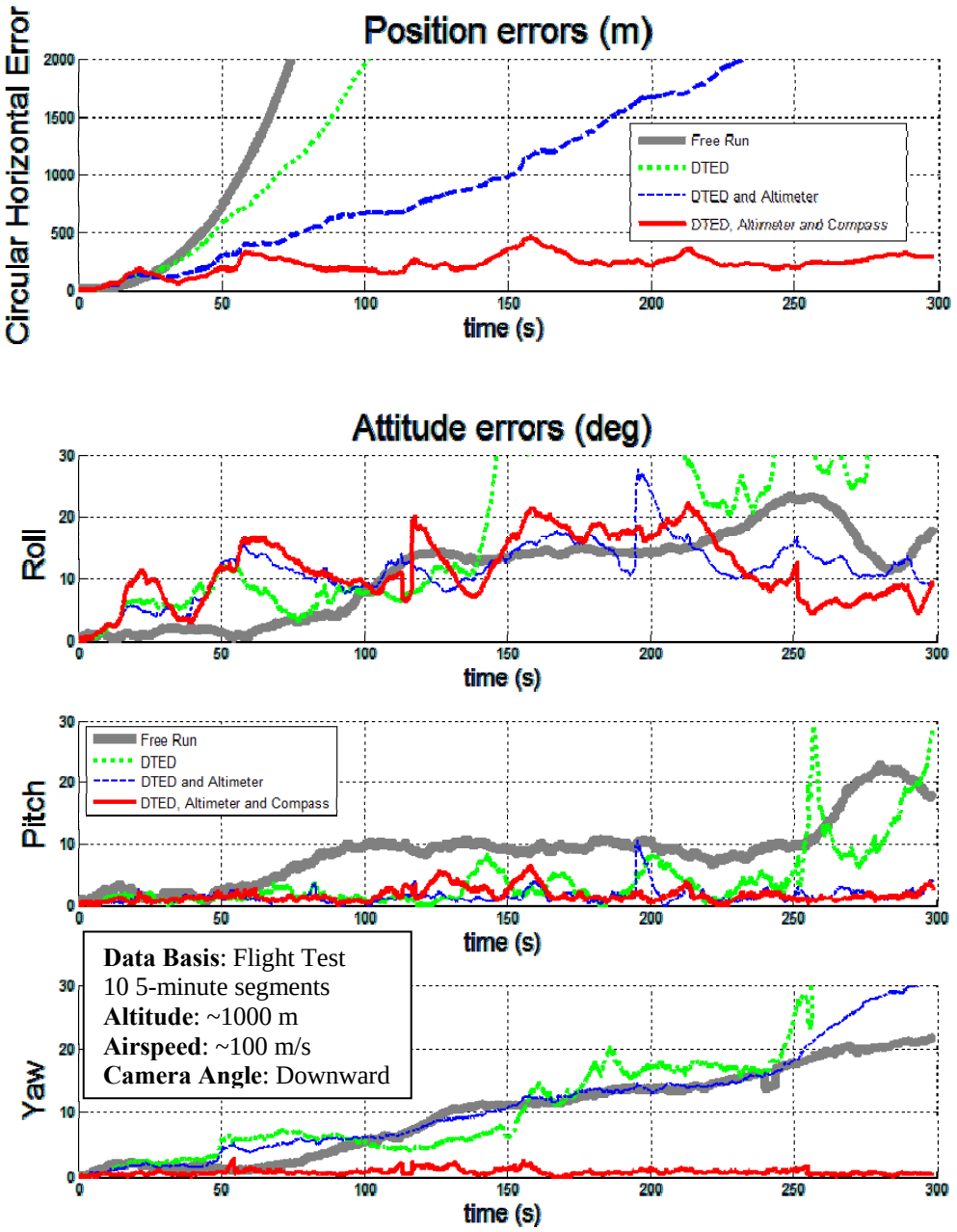

Figure 4.26: Effect of additional, non-image measurements.

For these flight test results, altitude data was taken from the Data Acquisition System (DAS) on board the test aircraft. The atmospheric bias (difference between barometric and geodetic altitude) was removed and assumed to be constant for the short

duration of flight. A longer duration flight would require a modeling and estimation of this bias. The heading measurements were not available on the DAS and were thus simulated. The heading measurements were modeled and simulated by adding white Gaussian noise to the TSPI values (1 degree standard deviation). Real data would require the modeling of the Earth's magnetic variation and installation errors, but no bias estimation.

The benefit of progressively augmenting the measurement equation is very evident. Tracking 10 LOO alone provides a marginal advantage over free inertial performance. In fact, the position errors grow exponentially, as did the free inertial errors. The marginal advantage is likely due to the ambiguity between pitch and velocity being resolved by DTED constraining landmark elevations (pitch errors are significantly improved). Adding altitude measurements to the equation makes a significant improvement, as the positional errors now grow linearly. The additional elevation constraint on the aircraft further resolves ambiguities in velocity, height and pitch as predicted. Notably, nothing done to this point has improved the heading error (yaw) over the free inertial condition. This demonstrates the need for an independent measurement of this state. Once heading is constrained with compass measurements, all navigation state errors are significantly reduced (almost leveling off).

If available, DTED, altimeter, and compass data should be incorporated into the measurement equation. They are all passive in nature and typically available on conventional aircraft. These Monte Carlo results indicate that all are needed to provide a viable navigation solution.

It is important to note that conventional INS also have difficulties resolving altitude and heading, and often require similar aiding strategies. The nature of gravity (decreases with altitude) causes an unstable altitude channel in INS mechanization. This is due to a positive gradient with respect to altitude errors when gravity, a function of altitude, is removed to resolve vertical acceleration. Many INS systems use a barometric altimeter or GPS to constrain the altitude. Heading estimation suffers from a different, but equally challenging nature. The yaw axis (z-axis in the *n-frame*) is essentially parallel to the local gravity vector, making the body frame gravity vector independent of heading. This requires a lengthy gyro compassing event during alignment on the ground, and often times, slaving the heading to a magnetic compass. This discussion is meant to demonstrate that the need for altitude and heading measurements is not unique to the system under test, or to SLAM-aided INS in general.

4.5.6 Effect of Number of Landmark Tracks. It was proposed in Chapter 3 that tracking many landmarks of opportunity can dilute the effects of having less than full observability. It has already been shown that tracking many surveyed landmarks improves performance by virtue of improving overall GDOP. Figure 4.27 illustrates the benefit of tracking many LOO by displaying the results of 4 Monte Carlo runs. Each run treats the same set of 10 5-minute segments with a different number of landmark tracks. All use DTED, altimeter measurements and the downward looking camera, but the behavior in Figure 4.27 is consistent for all conditions and combinations. The mean plus 2 sigma spherical position and velocity errors are displayed as well as each attitude state.

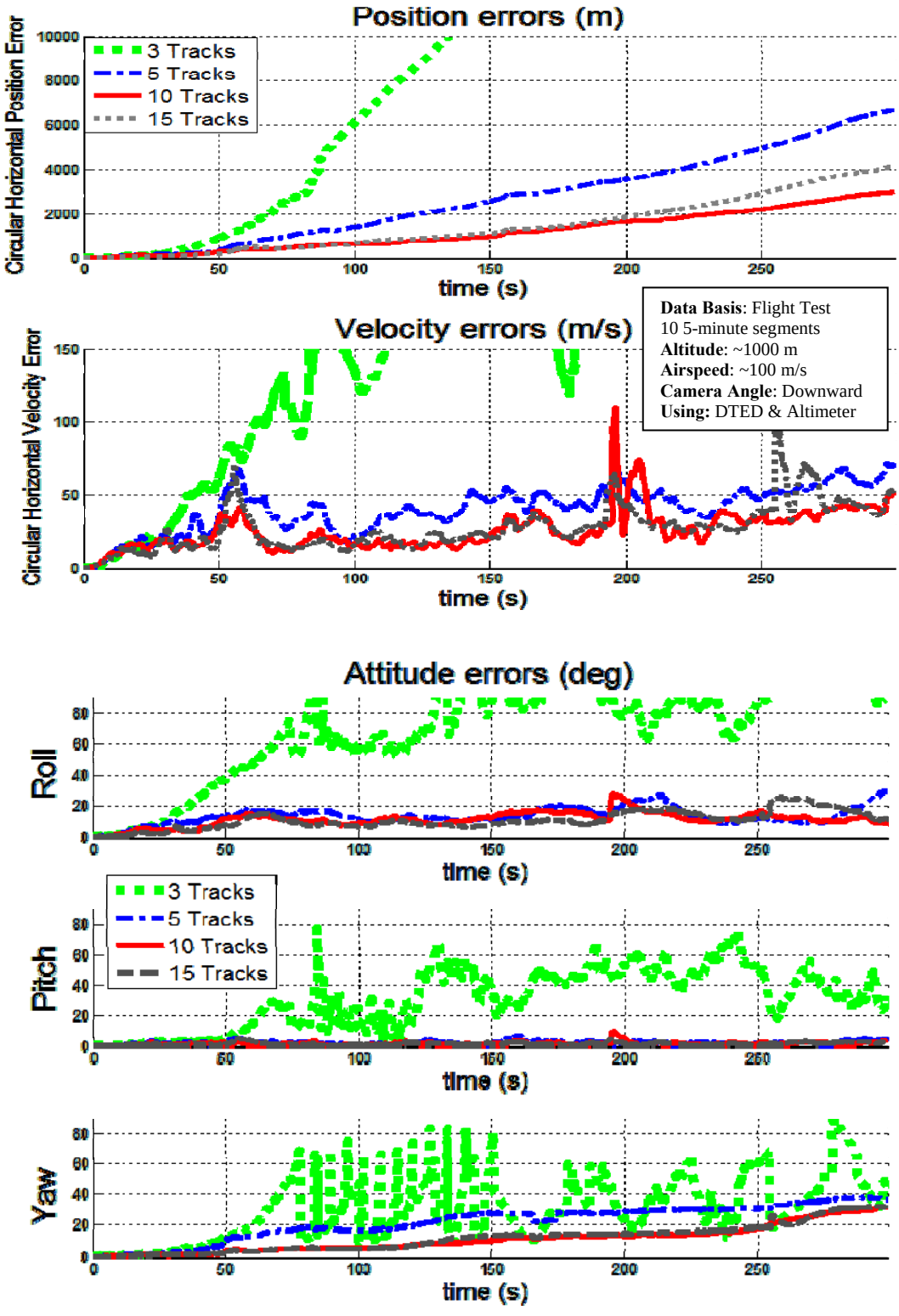

Figure 4.27: Effect of tracking many landmarks.

The benefit of many tracks is immediately evident. As the number increases from 3 to 10, all navigation states benefit. Tracking 3 landmarks gives very poor performance, but merely increasing to 5 gives orders of magnitude benefit. Pitch errors immediately drop to 1-2 degrees, and see little improvement when more tracks are added. Roll errors follow this behavior, but level out near 10-12 degrees. Position, velocity, and heading (yaw) incrementally improve as the number tracks grow to 10. Notably, performance starts to suffer as the number of tracks reaches 15. This was not predicted or expected, but could be due to many un-modeled effects. In either case, the number of tracks definitely affects navigation performance, and must be considered. It did turn out though, that 10 tracks was a very practical compromise between computational load and performance.

4.5.7 Effect of Camera Look Angle. It was proposed in Chapter 3 that the order and flow of landmarks through the scene effects the aiding action. The flow is ultimately driven by the look angle of the camera. Figures 4.16-4.18 showed that this is the case when surveyed landmarks are being tracked. However, because the position of the landmarks is already known, that case is a question of GDOP only. When landmarks of opportunity are being tracked, they must be located before they can provide aiding benefit. Thus, the rearward looking camera condition is expected to perform better than the forward.

Figure 4.28 illustrates the results of 5 Monte Carlo runs. Three of the runs treat a similar set of 10 5-minute segments with a different camera look angle, all using DTED, altimeter and heading measurements. Two additional runs show the effects of not using

heading measurements when looking down and aft. This case is not included for the forward looking camera, as it proved completely unstable. The 2 sigma spherical (2 standard deviation) position and velocity errors are displayed as well as each attitude state.

The predicted behavior is evident immediately. The forward looking condition proves completely unviable as all navigation states diverge quickly. The rearward and downward looking conditions both demonstrate relatively small and stable errors, each with its own advantage. The downward looking condition shows a 25-50% better position error, but suffers in velocity and attitude. The opposite is true for the rearward looking condition. This is likely due to the rearward looking camera capturing the feature near the horizon, giving better observability in the attitude states. The downward looking camera has more evenly distributed geometry and *V over h* throughout the scene. This gives better triangulation of landmark positions, and ultimately aircraft position, but lacks the dwell time near the horizon.

The runs without the use of a compass for heading measurements are included to illustrate the importance of this measurement and its independence from camera look angle. In both cases, the yaw angle errors grow roughly linearly and at nearly the same rate. Despite being able to track features on the horizon, the rearward looking camera condition requires a heading measurement just as much as any other. This is likely due to the small errors induced every time a landmark is lost and replaced. This Brownian-like motion occurs despite good attitude observability (while the landmark is tracked). A small error *step* is incurred with every new landmark track.

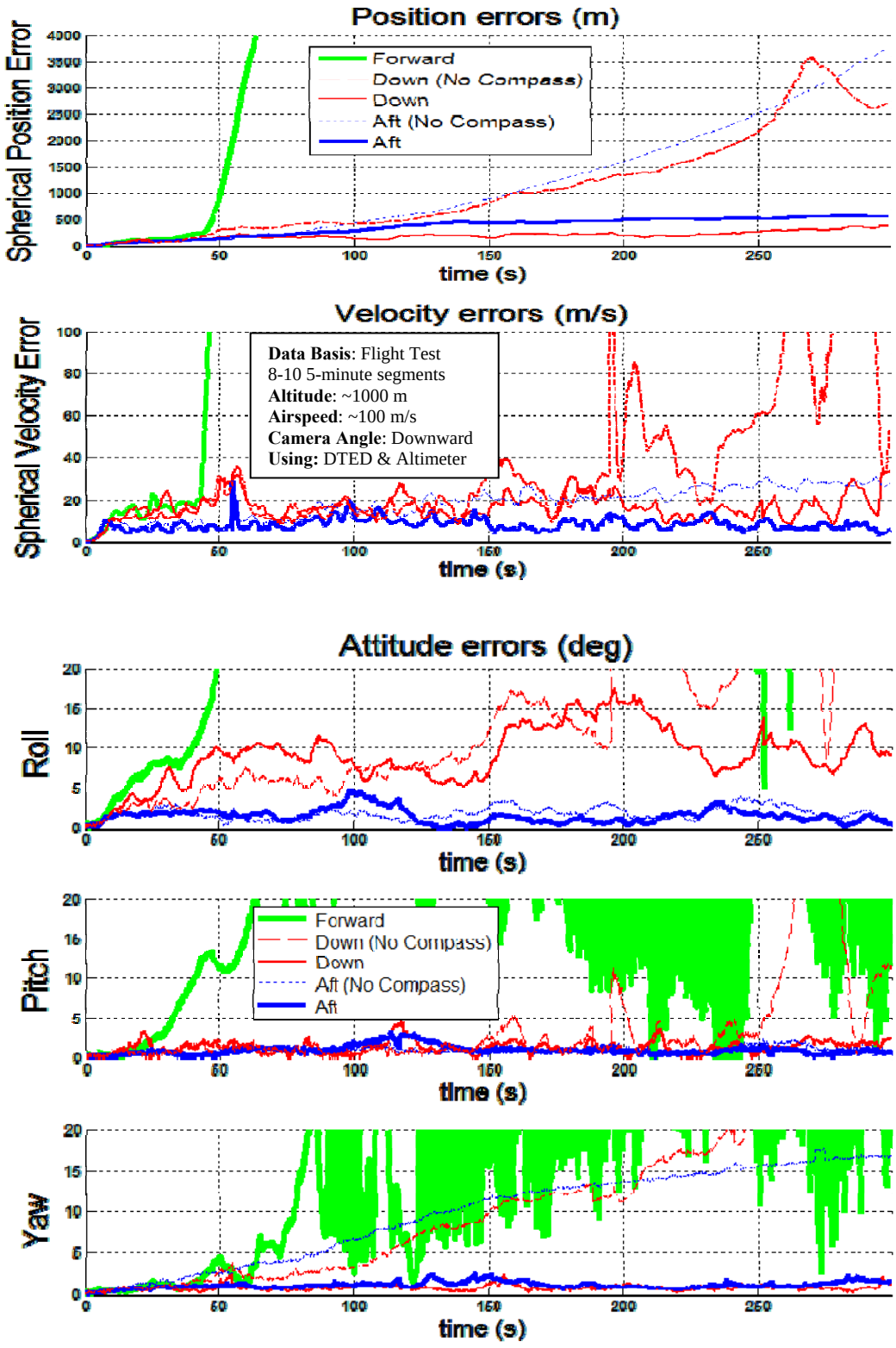

Figure 4.28: Effect of camera look angle.

4.5.8 Effect of V over h. It was shown in Chapter 3 that the apparent angular rate of a tracked landmark, driven by the *V over h* ratio, affects the aiding strength provided to the INS. This was supported in Section 4.5.3, as *V over h* affects the quality of the landmark map generated by a self-survey.

Another, more practical way that *V over h* affects INS aiding performance lies in the feature generation process. Any moving scene is ultimately blurred and distorted by a still camera. The blur is a function of shutter speed, detector sensitivity and ambient light (among other things). The feature generator must give consistent descriptors to the features, allowing them to be correctly and consistently identified frame to frame. Blurring an image changes the features, thus complicating the issue. If the image is blurred enough, the landmarks will not be consistently tracked, and dwell time reduced. Additionally, blurring a feature can elongate it. This calls into question the accuracy of the placement of a feature onto a landmark in the image.

For a downward looking camera, these blurring effects are generally a function of the *V over h* ratio at nadir. The scene is flat and blurring effects, though still a problem, are consistent throughout the scene (though a feature is blurred, the blur is doesn't change across the scene and the feature descriptors are consistent). If the camera is angled toward a horizon, the blurring effects are no longer consistent, but a function of position in the image. More blurring occurs where the angular rates are higher, further complicating the feature tracking problem.

During flight test, the camera shutter speed and gain were controllable to a point. For a given altitude, sun angle, and camera look angle, the camera could be set to

eliminate blur, at least to the naked eye. This held true until *V over h* exceeded 1-2 sec^{-1} and was significantly worse for the forward and aft looking cameras. At these conditions, the image was visibly blurry, and feature matches were very frequently dropped; aiding the INS was not possible. This became the major limitation of the system under test. Based on theory and simulation, a faster frame rate, shutter speed, and more reliable feature generator are predicted to improve performance at conditions with a *V over h* greater than 1.

For *V over h* conditions less than, but near 1 sec^{-1}, the image was not visibly blurry, but feature match performance was significantly degraded. It is unclear whether the feature generator was affected by unperceivable blur. Thus, it is difficult to separate the practical effects of the feature generator from the dynamic effects of the apparent angular rate. In either case, performance of the SLAAMR algorithm suffered as *V over h* increased from 0.1-1.0 sec^{-1}. Figure 4.29 illustrates 3 Monte Carlo runs at different *V over h* conditions. Each run consisted of 8-10 5-minute segments, and the significant mean plus 2 sigma errors (2 standard deviations) are reported. To eliminate the added blurring effects of the tilted camera, the downward looking camera was used for all three runs. The aircraft was flown at a constant ground speed of 100 m/s, and *V over h* varied by flying at different altitudes above the ground (1000-1300 m, 400-500 m, and 100-130 m).

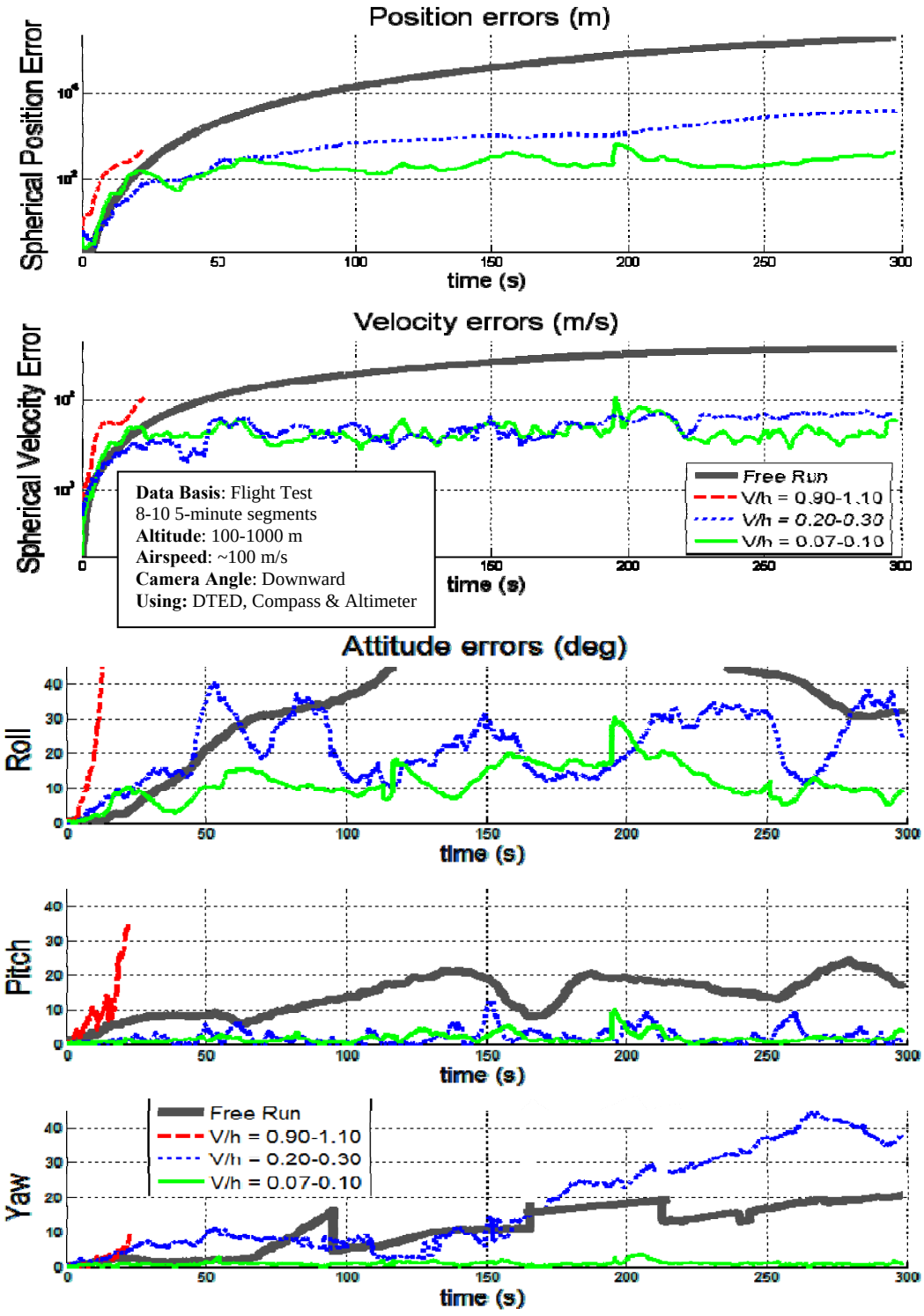

Figure 4.29: Effects of V over h on navigation performance.

As discussed before, the lower altitude runs could not be performed over the urban environment, but were flown over the Mojave Desert. The features were different in nature, but plenty (400-500) were available in every image. The influence of the different types of features is unclear, but a definite *V over h* dependence is evident. Also, the contribution of the specific system under test cannot be divorced from the results. For all the reasons previously mentioned, the apparent angular rate, or *V over h*, is a considerable factor affecting the performance of a SLAM-based INS aiding strategy.

The dependence on the *V over h* ratio is evident. Navigation performance increases as the *V over h* ratio decreases. The potential GDOP benefit above 1 sec^{-1} suggested in Section 3.3.2 was not evaluated, as even the 1 sec^{-1} case was overcome by image processing weaknesses. In fact, as is clear by the flight test data, the 1 sec^{-1} condition performed worse than the free inertial condition. This is likely due to the destabilizing effects of the false positive feature matches that tended to occur. The performance increase from 0.2 sec^{-1} to 0.07 sec^{-1} also supports the theory proposed in Section 3.3.2, but heading and position states benefit the most. At these low *V over h* ratios, most all of the feature matching problems experiences at 1 sec^{-1} were not evident, allowing the test data to follow theory.

4.5.8.1 Scalability. In Chapter 3, it was suggested that there is a scalability factor when considering aiding action. The observability grammian was scaled by the inverse of height squared. This implied that, as the absolute value of height decreased, the aiding action would increase. This is true despite the ratio of *V over h*. It was supposed that an aircraft flying low to the ground and very slow would achieve much

better aiding action. This case was not tested experimentally, but a simulation was. The aircraft was replaced by a soldier with a camera on his helmet. Instead of flying, the soldier was simulated to be walking with at *V over h* ratio of 1 (sec^{-1}); a 2 m tall soldier walking 2 m/s. The camera was pointed backward and a simulated sway in the soldiers step was simulated; see Figure 4.30.

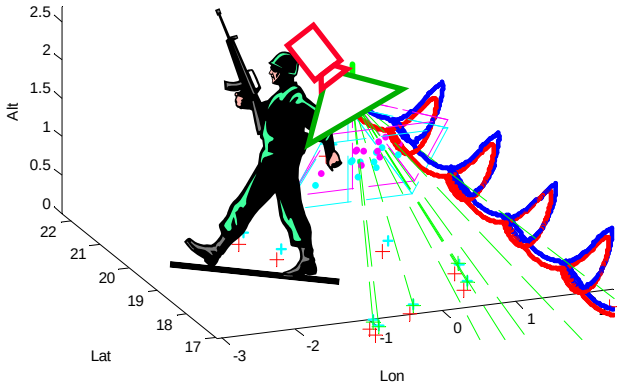

Figure 4.30: Soldier based SLAAMR system.

A Monte Carlo run of 30 5-minute segments was accomplished using simulated images as described before. No aiding was used other than tracking landmarks of opportunity. The results are displayed in Figure 4.31, and validate, at least in simulation, that the scalability principle is true. Position errors are on the order of the height of the camera, but attitude errors are on the order of the camera resolution. This is to be expected. Bringing landmarks closer to the camera does nothing for angular resolution, but, the projection of angular uncertainty onto a plane becomes much smaller. Thus, position and velocity errors are reduced.

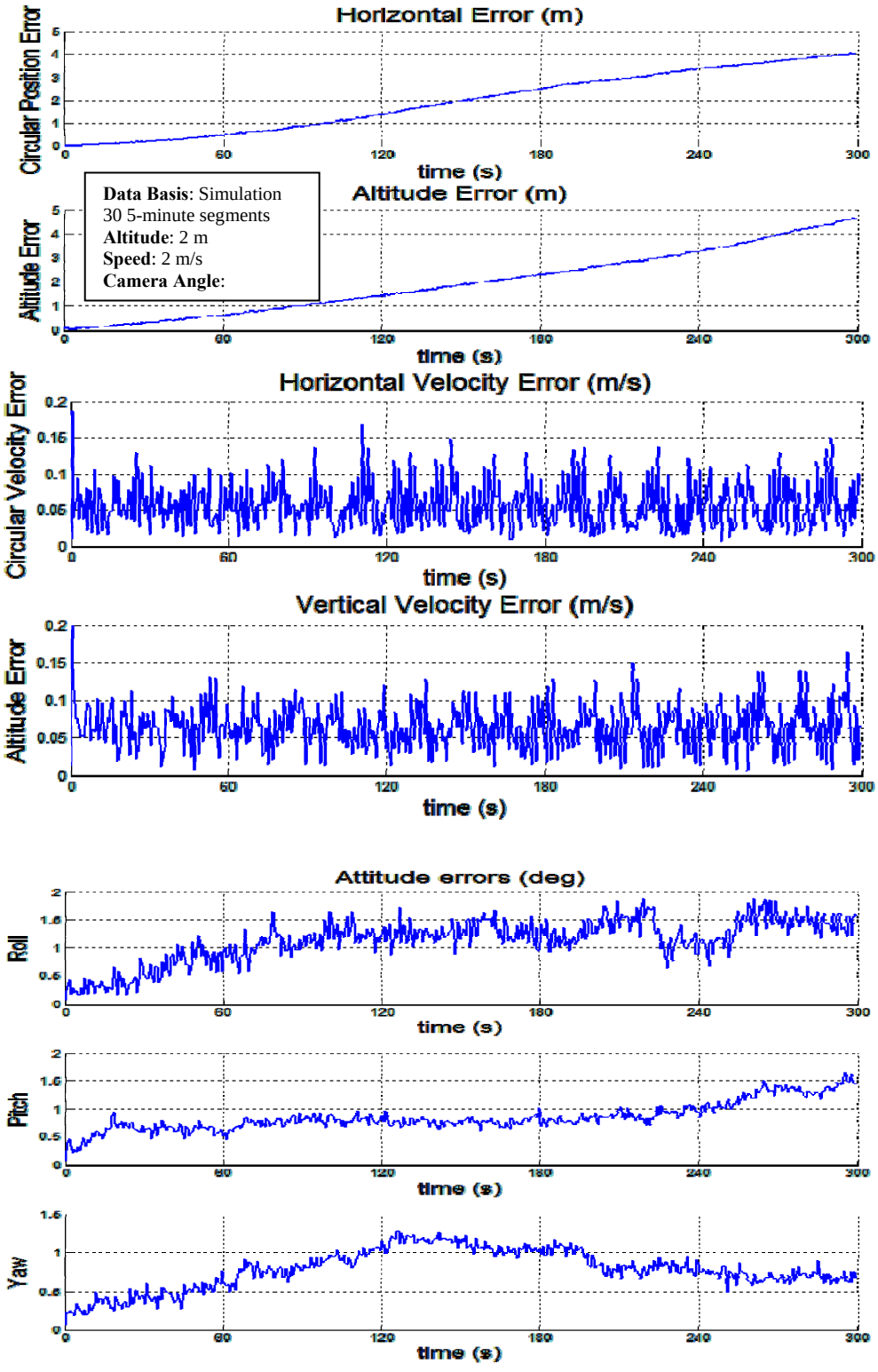

Figure 4.31: Simulated soldier based SLAAMR performance.

This simulation represents the best case scenario, as none of the complicating factors discussed up until now were simulated. This discussion was included merely to illustrate that both the *V over h* ratio and height itself matter. Also, this study supports Veth's work [26], and it suggests an area for future research using monocular vision.

4.5.9 Partial Survey Tracking. Suppose that a small set of surveyed landmarks were all that was available to, or could be practically carried by, a system. This partial survey or reduced catalog scenario accounts for practical limitations on third party survey capabilities and onboard computational resources. It was shown that precise navigation can be achieved by tracking 2-3 surveyed landmarks, and that reasonable navigation performance can be achieved (for short periods of time) by tracking landmarks of opportunity. If pairs or triplets of surveyed landmarks were placed along the intended flight path, the aircraft could navigate using LOO until the surveyed landmarks come into view. Tracking the precise, independently surveyed landmarks would eliminate the drift incurred to this point, and the process could repeat. Many conventional aircraft employing navigation grade INS use this strategy. Though the drift rates are slower and the spacing between surveyed landmarks (or position update points) is much greater than what would be required by the SLAAMR algorithm, the concept is sound and proven.

Two challenges are posed with a partial survey: recognizing the landmark when it is in view, and knowing where to look after the aircraft has drifted. The first challenge must be overcome by perfecting the third party feature generation issue. A third party must be able to survey a landmark and assign it a descriptor that is recognizable by the

aircraft from the expected viewing perspective. This problem is assumed to be solved for this study by using the images taken by the ownship cameras to generate the descriptors.

The second challenge is solved by the SLAAMR algorithm by its very design. Features being tracked are done so by estimating their projected position in the image (and associated uncertainty ellipse) using landmark and aircraft positions, and the measurement equation. A match is only considered if a candidate feature falls inside the uncertainty ellipse. The same is done for surveyed landmark searches. The uncertainty in the landmark's absolute position is determined by the third party, but the relative uncertainty is the sum of both the landmarks survey and the aircraft's positional uncertainty. This allows a wider search area as the aircraft drifts. If a match to a cataloged surveyed landmark is made, it is inserted into a track, overriding a LOO. The rest of the mechanization is identical to LOO tracking.

Figure 4.32 illustrates two Monte Carlo runs in which 2 surveyed landmarks are available every minute for 10 5-minute segments. This equated to a pair of surveyed landmarks every 6 km. The aircraft was flying between 1000-1300 m at approximately 100 m/s, and the camera used is looking downward, tracking 10 LOO at a time. DTED and altitude measurements are used as well. The performance is compared to the case in which all landmarks are surveyed and the case in which none are surveyed.

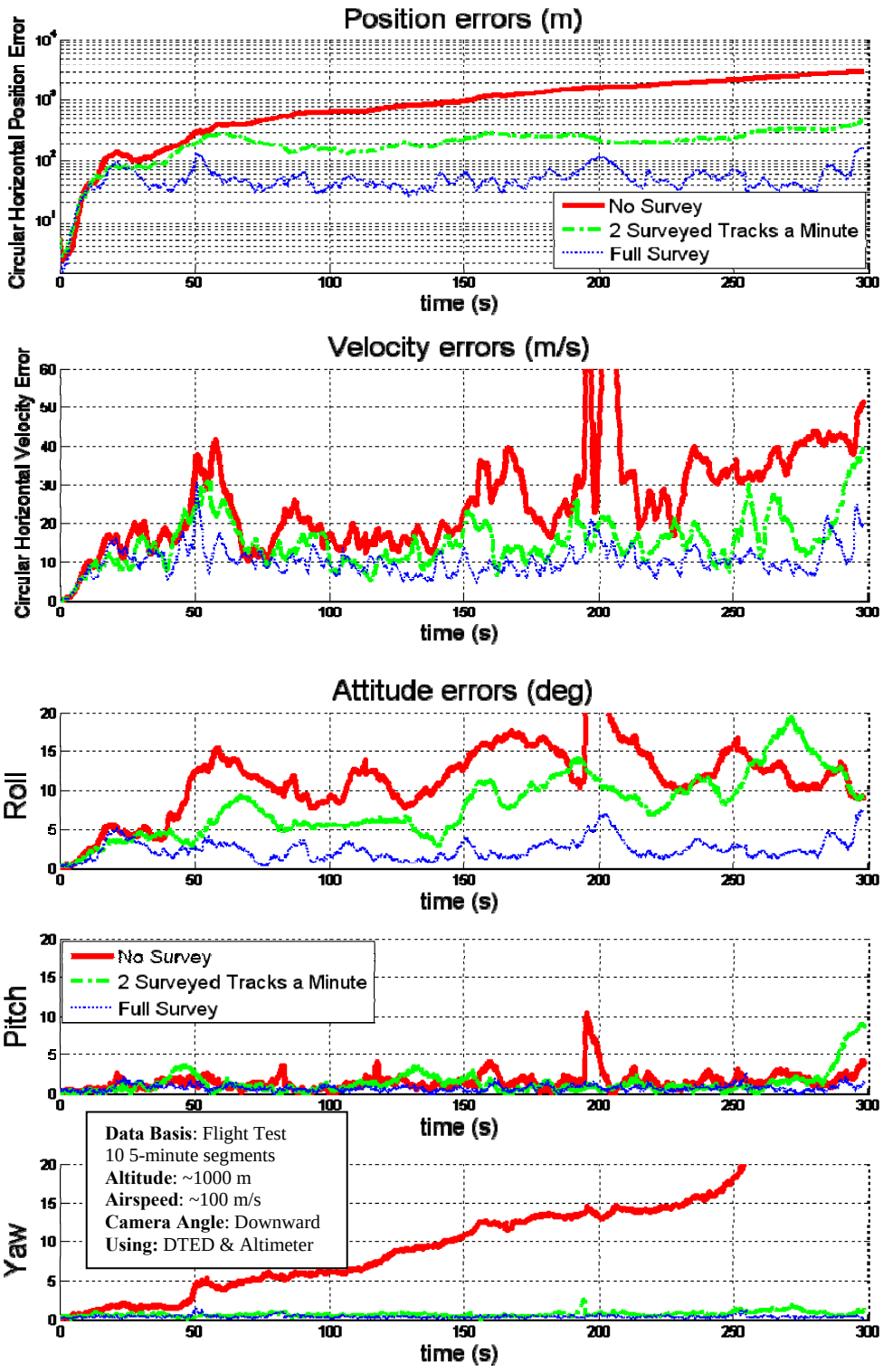

Figure 4.32: Partial survey navigation performance.

Using this hybrid method of tracking both surveyed landmarks and landmarks of opportunity proves to be a valuable compromise between navigation performance and a priori survey work. An aircraft can navigate a predetermined path with a small catalog of surveyed landmarks, lowering the computational burden of searching the catalog for matches, and reducing the need for an extensive survey by a third party.

Landmarks of opportunity near the surveyed landmarks will naturally receive a better self-survey by the aircraft. If the route is flown often enough, this effect will be felt further out, eventually creating a precise, self-surveyed map. Thus is the nature of a Kalman filter; all measurements taken are used, refining the estimate.

The ability to get by on a small number of occasional surveyed landmarks allows for imperfect feature recognition. If a large catalog of surveyed landmarks is carried, the algorithm need only recognize a few, occasionally. Care would need to be taken to avoid false positive matches, as they would destabilize the navigation solution.

4.5.10 Landing Profile Performance. The landing profile was flown as a demonstration of concept. The goal was to explore the military utility of using the SLAAMR algorithm for an autonomous landing system or precision approach guidance system. The camera that was used in the test photographed in the visible light spectrum, but without loss of generality, an infrared or similar type of camera could be used at night or in the weather. Figure 4.33 displays three Monte Carlo runs; flight test results using surveyed landmarks, flight test results with feature matching errors eliminated, and simulated data. Each run consisted of 10 straight-in approaches to landing that followed a 3 degree glide path and lasted 73 seconds. The flare (defined by 50 feet above the

runway) occurred at approximately 70 seconds. Up to 10 surveyed landmarks were tracked at any given time, and the forward-looking camera was used. Surveyed landmarks were chosen over landmarks of opportunity for two reasons. The navigation estimate had to be precise for a landing task and feedback control. Also, it is reasonable to expect that the runway environment could be well surveyed with many landmarks. The descriptors could also be generated (by a third party) from the same perspective as they would be viewed by the test aircraft. The Monte Carlo plot contains the individual event spherical errors in position, velocity and attitude, and the mean plus 2 sigma errors are displayed (2 standard deviations).

In general, the performance of the SLAAMR algorithm was not precise enough to be used as a landing guidance and navigation system. Positional and velocity errors were acceptable for most of the profile. A 20-30 meter spherical position error and a 5-10 m/s spherical velocity error are adequate for navigating down the glide slope. However, after approximately 60 seconds (13 seconds from touchdown, and 60 meters above the ground), the errors began to grow geometrically, reaching 160 meters in position and 30 m/s in velocity by touchdown. At this point in the profile, the combination of high *V over h*, the low angle scenery and image blur overcame the benefit of the surveyed landmarks. Landmarks currently being tracked were dropped after a single frame, and potential surveyed landmarks not tracked were not recognized. Essentially, the system was running free inertial, gaining no aiding action from landmark tracks. This was exacerbated by destabilizing false positive matches.

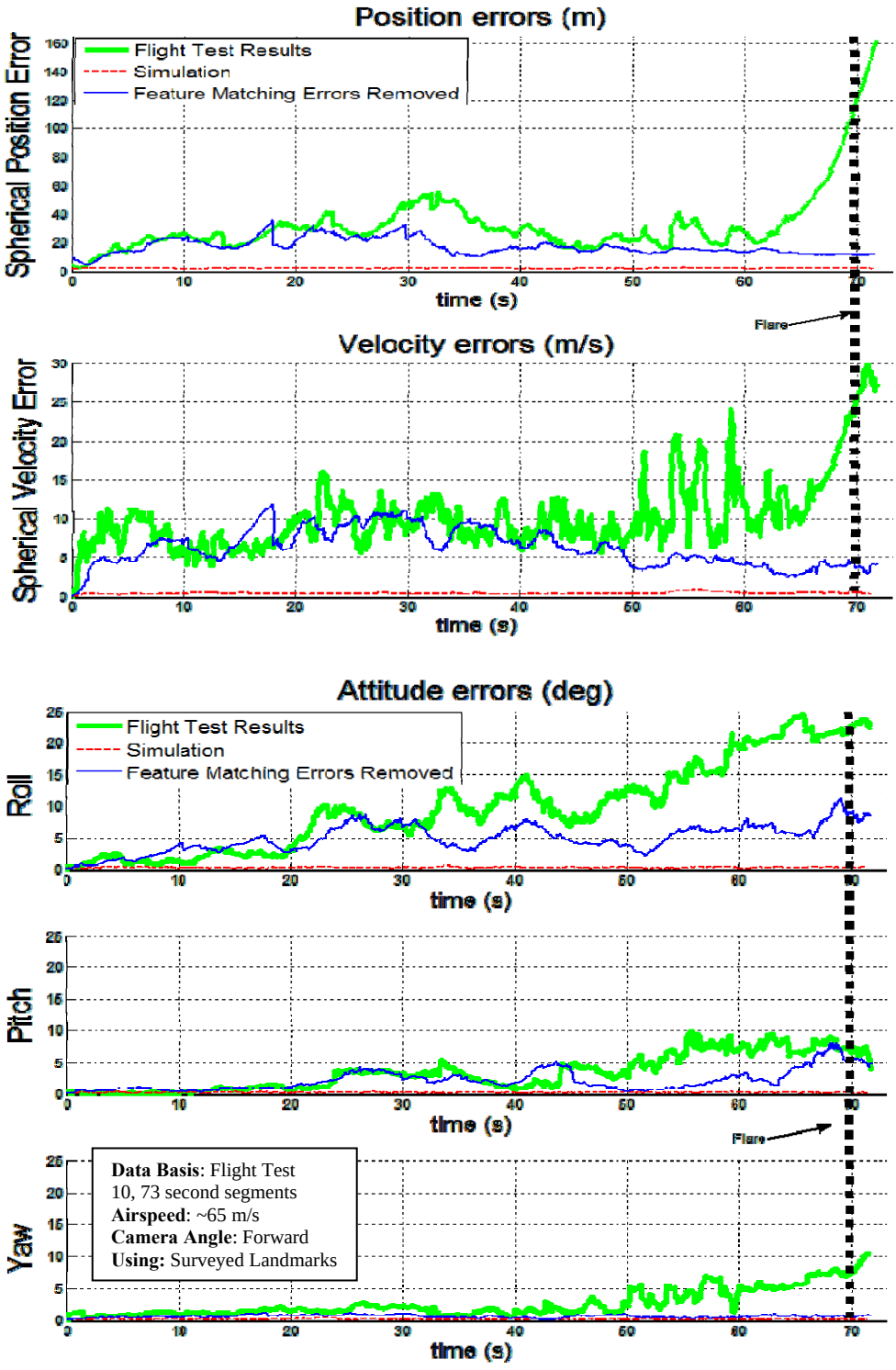

Figure 4.33: Landing profile performance using surveyed landmarks.

Not finding the surveyed landmarks caused great instability at a critical time in the profile. To examine the impact, TSPI and DTED were used to eliminate some of the feature matching errors by allowing the SLAAMR algorithm to find the surveyed landmarks in the first place. However, this did not aid frame-to-frame tracking. Because the surveyed landmark position is essentially known, aiding can be achieved with a single frame/measurement. Figure 4.33 shows that correcting the feature matching errors eliminates the free inertial drift that occurs after 60 seconds (described above). The position and velocity errors at touchdown became acceptable for a landing task on a conventional runway (15 meters and 5 m/s respectively). The vertical channel (altitude and vertical velocity) is the most critical when the aircraft is close to the ground. A sink rate greater than 1-2 m/s is considered a hard landing, so the vertical velocity uncertainty must be at least that precise below the 2-3 sigma altitude error. The flight test results do not guarantee this constraint, but it could be accomplished with altimeter aiding or a laser height finder that is common on many UAVs.

Landmark tracking frame to frame was greatly impeded by the nature of the scene. The scene was at a very low angle (3 degrees) and moving very quickly near the bottom. Blur and scale greatly complicated the matching problem, and feature persistence dropped considerably. The features grew significantly in scale as they translated the scene. The descriptors changed, reducing match probability, and the associated coordinates were not consistently placed on the same spot in 3-dimensional space. At these conditions, a small pixel position error, translates into a large distance error in 3-dimensional space. The primary cause poor matching was that the flight path

was relatively parallel to the Earth. A set of purpose-built landmarks on a vertical surface could mitigate these problems. This suggestion is supported by data taken up and away, looking down at a flat scene. In this case, matching features and landmarks was accomplished consistently. It is not unreasonable to place distinct, scale-invariant patterns (checkerboard, etc) around the airfield on vertical placards or small billboards. Airfields are already littered with markings and lights. However, such purpose-built landmarks were not part of this flight test.

Position errors exhibited relatively adequate performance, but the attitude states are another issue altogether. To guide and land an aircraft precisely, attitude state must be known and controlled to a degree *or less* (depending on the aircraft). Not only do the flight test results not come close, but they drift over time. Removing the matching errors corrects the heading state, but only reduces touchdown roll and pitch errors by approximately 60% and 10%, respectively. Even with the matching errors removed, the attitude errors are unacceptable for a landing task.

Simulation suggested a very a highly precise navigation solution as shown in Figure 4.33. The simulated results suggest that the benefits of scalability and surveyed landmarks overcome the geometric and dynamic penalty of the ever increasing *V over h* ratio. The difference between simulation and test data lies in the image processing and feature generation at high *V over h* ratios and at such a low altitude and descent angle. This is supported by the fact that navigation state errors and feature matching maintained an acceptable level of performance until an altitude threshold was crossed. The mathematics and simulation indicate that very precise navigation should be possible, but

the un-modeled effects on the imagery and feature tracking overcome this potential. It is thus proposed that improved feature generation and matching, coupled with finer optics and faster frame rates could improve performance of the SLAAMR algorithm allowing it to approach that suggested by simulation.

Overall, the test data showed poor performance, failing adequate performance criteria by significant margins. However, the data did show some promise for a SLAM-based algorithm like SLAAMR. In general, when matching errors were removed, the navigation state errors matched the predictions in nature and in sense, but were larger by orders of magnitude. Improving the feature generation/matching process could bring the system into line with simulation. These test results suggest that the SLAAMR algorithm could be used to provide precision landing navigation if the image processing is improved or the suggested strategies (purpose-built landmarks, etc) are employed.

4.5.11 Capstone Design – Long Duration Performance. The discussion up until this point has illustrated the great challenge posed by attempting to implement a SLAM navigation system. The SLAAMR algorithm is designed to maximize the aiding benefit of landmarks of opportunity by inserting them into the algorithm immediately, but this is not enough to provide long-term navigation stability. The great benefit of pointing the camera down or aft (versus forward) and using altimeter and heading measurements was shown. Without a great leap in feature generation/tracking technology, and significantly more precise optics, landmark of opportunity tracking alone will not provide a MEMS grade IMU the aiding it needs to be a viable navigation platform. However, adding the passive altitude and heading measurements, and using DTED to estimate landmark

position initially does provide a navigation solution comparable to much larger and more expensive navigation grade INS.

Figure 4.34 illustrates the results of using altimeter, heading and DTED during a 30 minute flight with a *V over h* of 0.08-0.10 sec^{-1}. This figure only displays the significant navigation states. Vertical position, velocity, and heading errors were small and un-noteworthy because they were constrained by the altimeter and compass measurements. The pitch errors were also very small, and identical in both cases. It is believed that the combination of the altimeter and DTED eliminated any ambiguities between pitch, forward velocity, and altitude (making these errors very small). Both the downward and aft looking results are displayed for comparison.

From a positional error standpoint, looking downward performs 3-4 times better than aft, reaching a maximum of 400 m of error in 30 minutes versus approximately 1000 m. In both cases, this is comparable to a navigation grade INS which could expect to drift 1 mile (1600 m) in one hour. This isn't the whole picture however. The aft-looking condition outperforms downward-looking in both roll and velocity by a factor of 2, with maximum errors at 6 degrees and 10 m/s versus 12 degrees and 20 m/s. The aft camera captures the horizon, providing better attitude aiding action. If these measures are more important than position, pointing the camera aft is optimal. However, both of these measures are outperformed by a navigation grade INS. The following figure (Figure 4.35) repeats the same scenario as above, except without the benefit of the heading measurements.

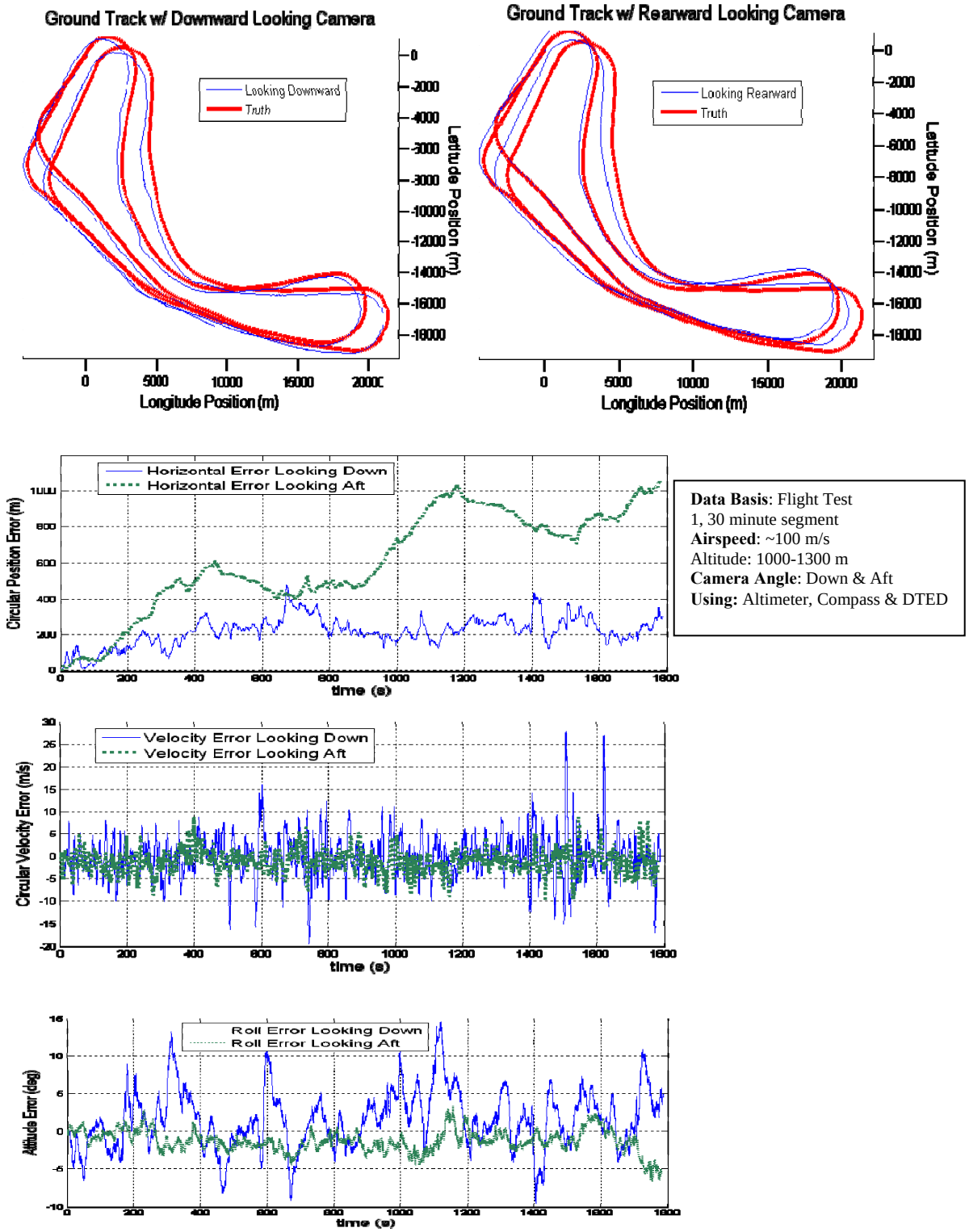

Figure 4.34: Long duration flight using altitude, heading and DTED measurements.

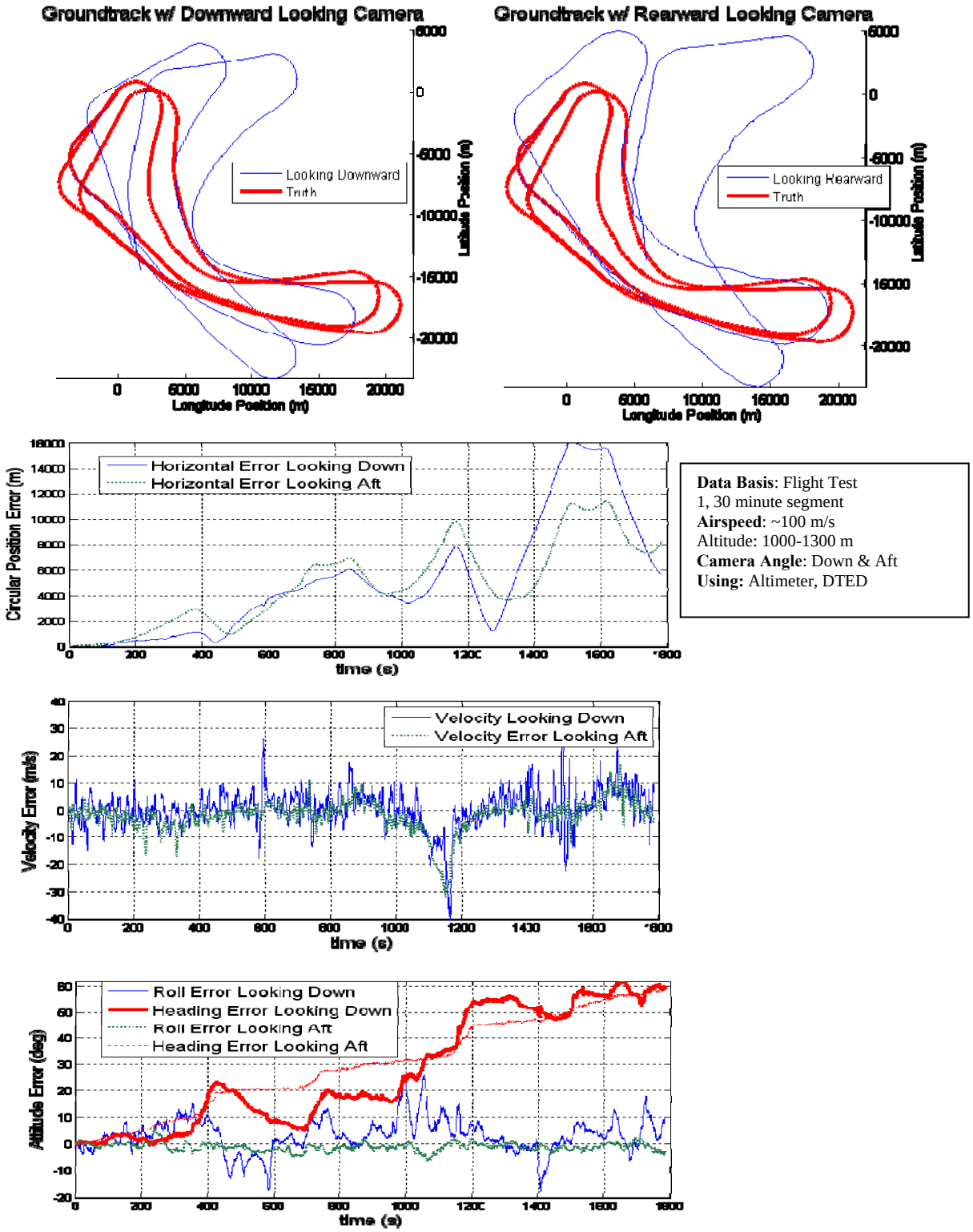

Figure 4.35: Long duration flight using altitude and DTED measurements; no heading measurements used.

The addition of the heading error (on the lower left attitude error plot) illustrates the dominating source of position error. Despite roughly equivalent velocity and attitude errors as Figure 4.34, the position drifts significantly farther. Examining the heading error and the ground track, it is evident that the drifting heading estimate points the otherwise good velocity estimate in the wrong direction. Since position is the integral of velocity (in INS mechanization), the miss-pointed velocity causes the position to drift away. It does not run off exponentially like in the free inertial case, the n-frame simply rotates as it goes.

Another important finding is made from this test. Despite best efforts (measuring altitude and incorporating DTED for landmark position estimation), the heading state appears un-aided, almost independent of the rest of the navigation state. The same behavior is seen despite the number of LOO tracked. The proposal that diluting un-observability with many tracks appears to break down in the heading dimension. The un-observability is not evenly distributed across all states, and at least one vector in the null space of the observation matrix is closely aligned with the heading dimension. In Chapter 3, it was shown that a measurement of any attitude state would have the same effect of increasing the rank of the observability grammian, but Figures 4.34 and 4.35 suggest that the heading measurement has a more significant impact on the grammian condition number. Ultimately, these findings illustrate the vital importance of independently measuring heading if a viable long-term navigation solution is desired.

4.5.12 Ad Hoc Techniques That Did Not Work. The SLAAMR algorithm is built around an extended Kalman filter that assumes certain properties of the system being

modeled and states being estimated. The EKF assumes that all errors (measurement and process driving noise) are zero mean and Gaussian, and that the linearization of the system is accomplished about a *good* nominal. Taking bearings measurements from images is inherently non-linear and plagued with errors violating these assumptions. Optical calibration errors are deterministic and not normally distributed. Feature generation errors are systematic, and not Gaussian. Any ZAPI navigation technique, such as the *Far Guess* initial landmark position estimation routine, inherently biases the initial landmark position estimate. The INS-derived nominal about which the system is linearized drifts over time. For these reasons, several ad hoc techniques were derived to mitigate the non-linear and un-modeled error sources. None significantly improved performance without an inordinate computational penalty. They are included for informational purposes and to strengthen the argument for using the basic SLAAMR algorithm described up to this point. The ad hoc techniques evaluated were:

- *Second Order Extended Kalman Filter*. This filter model provided an insignificant performance increase, but at a huge computational penalty [15]. This suggests that the main source of error in the measurement equation resides in the drifting nominal, not an insufficient error propagation model.
- *Partial Batch Estimation*. The first two frames in which a new landmark was visible were used to triangulate the initial position estimate. This technique proved no better than the base algorithm, and complicated computation.
- *Delayed Inclusion*. Landmarks with fewer tracking frames than a set threshold (2-5) were excluded from aiding the basic navigation states. This was

accomplished by zeroing out the applicable Kalman filter gain matrix elements and error state uncertainty cross-covariance components. This technique showed significant performance degradation and posed a computational penalty.

- *Predictive Feature Selection*. The scale, orientation and descriptor properties (provided by SIFT©) were used in an attempt to select features that would both persist and have stable placement characteristics. There was no perceived advantage to any of the properties, except scale. Large scaled features tended to be associated with low spatial frequency portions of an image, and were more resistant to aliasing and fading away. However, they also tended to be placed on the centroid of the feature and were inconsistently placed if the scale changed, as was the case with the forward and rearward cameras. Also, features near the edge of the image appeared to gain some measure of scale from image edge effects, and changed as they translated away from the edge. The best scaled features were in the middle of the pack, but as relative scale values changed with the scene, it was difficult to predict.
- *Tracking Coast Mode*. Landmarks with tracks that became lost were allowed to coast for a determined number of frames (2-5) before the track was considered broken. While in coast mode, the position estimate and uncertainty remained unchanged, and no measurements were taken until a positive match was made in later frames. If the coast time expired, the track was replaced with a new landmark. This mode showed slight performance degradation, and increased computational load.

- *Higher Quality IMU*. Though a higher quality IMU does significantly decrease the rate of INS drift, the high quality IMU may preclude any aiding strength from tracking landmarks of opportunity. Practically speaking, observable drift had to occur in the time that a feature could be maintained in view if it were to have a significant impact. For example, if the INS drifted 100 meters in an hour, tracking a landmark for 10 seconds with a real camera (limited resolution) did nothing as the drift over 10 seconds was unobservable. This would not be the case if an infinite resolution camera was used or landmarks could be maintained in view for very long periods of time. An INS using a high quality IMU did not benefit from tracking landmarks of opportunity in the scenarios described herein.

4.6 Summary

In summary, the flight test portion of this study generally validated the assertions and propositions made (in Chapter 3) that led to the principles of design. Image processing proved to be the most challenging aspect of the SLAAMR algorithm, and flight test data served to help characterize its performance. Though the data did not always match simulation precisely, it did serve to illustrate the effects of the major influencing factors (predicted in Chapter 3), and to characterize the true aiding potential of bearings-only measurements. The SLAAMR algorithm design is ultimately validated by navigation grade performance in the capstone evaluation.

V. Conclusions and Recommendations

This thesis developed and explored the fundamental observability of bearings-only measurements and their effect on aiding an inertial navigation system. This exploration was used to develop a practical, robust Simultaneous Location And Mapping (SLAM) algorithm, and is expressed as a collection of principles of design. An experimental apparatus was built and flight test data collected in a representative full-scale environment. The flight test data was collected to support and demonstrate the principles of design, and illustrate the challenges faced when operating in a real-world environment. This chapter summarizes the results of the flight test, and draws conclusions about the principles of design. Recommendations for future work will be made, based on shortcomings in this test and areas that showed promise.

5.1 Conclusions

Overall, the navigation performance of the Simultaneous Location Aiding And Mapping Recursively (SLAAMR) algorithm performed significantly worse with flight test data than in simulation. This was due to the many non-linear, non-Gaussian errors sources in image processing and optical effects that were not accounted for in the system model or simulation. Generally, however, the sense and nature of the navigation errors matched simulation, suggesting that the underlying model is sound. The study of the observability grammian matrix suggested various factors influencing performance. Flight test results validated the predictions, and each is addressed in this section. Moreover, the

flight test results were based on the statistics of many independent data runs, capturing the true nature and trends.

5.1.1 Surveyed Landmarks are the Practical Equivalent to Perfectly Known Landmarks. Perfectly known landmarks allow full navigation state observability with only two tracks. This was shown mathematically by examining the observability grammian matrix. Full observability ensures that the navigation state errors are bounded by the resolution of the measurement system and by geometry. It was asserted that, if a landmark is independently surveyed to a fine enough precision (not perfectly known), and included into the state vector for simultaneous estimation, the navigation state will receive practically the same aiding benefit as the known landmark case (despite the state vector dimension growth). Flight test data illustrated this effect plainly. The errors were bounded over time, and limited by the geometry of the tracked landmarks. Therefore, though the system never obtains full observability, independent but imperfectly surveyed landmarks are the practical equivalent to the perfectly known case. In fact, the quality of the survey merely drives the magnitude of the bounded navigation state errors, but any independent survey will constrain the drift.

5.1.2 Full Observability is Achievable by Tracking Only Two Landmarks. It was asserted that only two landmark tracks are required to achieve full observability (in the perfectly known landmark position model). This contradicts intuition, which suggests three are needed for triangulation. Flight test data supported the first assertion, as there was no fundamental difference between tracking two or three surveyed landmarks. (It was previously shown that surveyed landmarks are the practical equivalent to the perfectly known case.) Both showed bounded navigation state errors, consistent with full

observability. The three-track case showed smaller average error magnitudes, but this was determined to be due to better average GDOP and observability grammian matrix condition number, not a difference in the degree of observability (observability grammian matrix rank).

5.1.3 The SLAAMR Algorithm Naturally Builds a Surveyed Map of the Environment, but the Quality is Driven by Optics, Frame Rate, Tracking Angle and V-over-h Ratio. It was suggested that an extended Kalman filter based SLAM algorithm naturally builds a map of the surroundings and that this map could then be used to constrain errors in the navigation state. The strength of navigation state aiding is driven by the quality of the map generated. It was also asserted that the precision of the map position estimates approach that of the aircraft position estimate. This theoretical lower bound was found also to be a function of optical resolution, camera frame rate, landmark cumulative tracking angle, altitude (i.e., scalability) and the velocity-to-height ratio. The maps generated in flight test proved to be of very poor quality. This was, however, attributed to the test apparatus used in flight. The combination of poor camera resolution, low frame rates, and small tracking angles due to poor feature tracking, limited the resolution of the self-surveyed maps. However, despite poor performance, the data validated the model relating map precision to these factors, and the original assertions were supported. Precision self-surveyed maps can be generated, but a precision sensor system is required.

5.1.4 Navigation with Zero A Priori Information Requires a Much More Precise Sensor System than was Tested. It was asserted that a self-generated, self-surveyed map of landmarks can be used to constrain navigation state errors. This approach was

demonstrated in simulation and previous studies. However, due to the poor quality of the self-surveyed landmark maps generated by the flight test apparatus, zero a priori information navigation (using the *Far Guess* initial position estimate routine) performed worse than the free inertial case. A previous study showed that this type of navigation can prove to work in certain conditions, but was not representative of a realistic flight environment [5]. The full-scale and representative flight environment used in this test proved too challenging for the flight test apparatus. Thus, zero a priori information navigation cannot be validated in this study. Due to the poor precision test apparatus, it also cannot be invalidated.

5.1.5 Additional Passive Aiding Overcomes the Challenges Posed by Full-Scale, Realistic Flight Environments. It was shown that the additional passive measurements of altitude, heading, and use of DTED for initial landmark position significantly increase system observability and resolve ambiguities in bearings-only measurements. The realistic flight test environment and scale posed great challenges for the SLAAMR algorithm, and errors quickly became unacceptable for flight. Landmark tracking was limited by feature generation, and the scalability principle meant that what were small errors at low altitudes (seen in previous studies), were very large at realistic altitudes (used in this test). The additional measurements of altitude, heading and DTED significantly improved navigation performance, making up for poor image processing. Each was shown to be critical for long-term stability, and these dimensions are weakly observable from bearings-only measurements. They also follow the low-cost, jamming-resistant, stealth model, since these measurements are readily available on many aircraft and are passive in nature. Image processing will always be the crux of a SLAM-based

algorithm. Measuring heading, altitude, and incorporating DTED significantly make up for poor image processing by complementing bearing-only measurements in unobservable dimensions.

5.1.6 Tracking Many Landmarks Ensures Good GDOP and Improves Performance. It was proposed that tracking many landmarks of opportunity would improve performance by diluting the degree of un-observability, guaranteeing better average GDOP, and ensuring at least two well (self) surveyed landmarks. Flight test data clearly showed that performance improved as the number of tracks increased to ten, supporting this assertion. There was a slight and unexplained decrease in performance above ten, but ten proved to be enough and did not induce too heavy a computational burden.

5.1.7 Camera Look Angle Significantly Affects Navigation Performance. It was proposed that the direction in which the camera (used for imaging the environment) is pointed, greatly affects the order and flow of geometry in a scene. This in turn affects the aiding action provided to the INS. Flight test data clearly showed that camera look angle plays a significant role in the performance of a SLAM-based system. Looking forward to the horizon performed very poorly, as expected. Looking downward or rearward performed significantly better than looking forward (orders of magnitude by some measures), each having its strong points. Rearward performed better in attitude and velocity states and downward in position. The rearward condition was expected to perform better than downward in all cases, but it is suspected that the more complicated image processing problem looking rearward leveled the playing field. The downward

looking camera was presented a benign, flat scene with little distortion or blur, while angling the camera created feature scale changes and depth of field problems.

5.1.8 Velocity-to-Height Ratio (Feature Angular Rate) Significantly Affects Navigation Performance. It was suggested that the ratio of velocity to height (*V over h*), or more specifically the angular rate of features as they pass through a scene, drives navigation aiding effectiveness. The effect on performance is manifested in two ways, the dynamics of the observability grammian and the ability of the feature generator to place features consistently on landmarks. Dynamically, a lower *V over h* ratio presumably benefits the attitude states, while a higher ratio benefits positional states. There exist *sweet spots* where aiding is most evenly distributed, but which are difficult to predict. Optimally, a system should attempt to capture landmarks exhibiting the span *V over h* ratios in order to maximize aiding benefit. Practically, however, the image processing was the driving force on performance. Flight test data showed that lower *V over h* ratios performed better than high, and a *V over h* near 1 sec^{-1} performed worse that free inertial. At this condition, landmark tracking became nearly impossible due to poor image processing, while *V over h* between 0.30-0.05 sec^{-1} was not terribly affected. Image processing is a function of the specific system under test, but will challenge any SLAM-based system. In this case, any potential benefit of good GDOP at higher *V over h* values was overcome by poor image processing. It must be noted that the observability study accomplished in Chapter 3 focused on the deterministic measurement equation, neglecting the contribution of the inertial measurement unit quality. The poor quality IMU (used in test) may have impacted the algorithmic performance in a way not captured by this analysis.

5.1.9 Using a Partial Catalog of Surveyed Landmarks Provides a Practical Compromise to A Full Survey. The SLAAMR algorithm allows the seamless insertion of surveyed landmark tracks into a population of landmarks of opportunity. This enables a smaller set of surveyed landmarks to be carried in memory and accounts for poor surveyed landmark recognition. Both realistic scenarios gain the benefit from surveyed landmarks when available (correcting drift), and can continue navigating when they are not using Landmarks of Opportunity. Flight test data showed that a minimal set (2 every minute or 6 km) provided a very good compromise for this configuration, as navigation state errors were bounded and reasonable for flight.

5.1.10 The SLAAMR Algorithm Shows Promise for Use as a Precision Landing System, but the System Under Test Has Major Deficiencies. The landing profile was flown as a proof of concept test for military utility in precision landing tasks. Simulation and theory suggested that highly precise navigation could be achieved during landing. Flight test data illustrated that the real world image processing problem presented by such a task was not solved by the specific system being tested. However, the (artificial) removal of some of the many complicating feature tracking errors showed great improvement. It is thus proposed that a more refined system that solves the image processing problem could prove viable for this task.

5.1.11 The SLAAMR Algorithm Provides Navigation-Grade INS Performance for Long-Term, Up-and-Away Flight When Tracking Landmarks of Opportunity. When the SLAAMR algorithm is employed fully, to include altitude, heading, and DTED data, the navigation performance was on the order of a contemporary navigation-grade INS. The addition of the altitude, heading and DTED data overcomes the challenges presented by

image processing and scaling. If flown high enough to provide a V over h of approximately 0.01 sec^{-1}, and pointing the camera downward or rearward, the system under test proved viable for use in a contemporary aircraft on a realistic mission. The problem of scalability to realistic altitudes and airspeeds was also solved, as well as allowing long distance travel into unknown environments populated with unknown types of features. Improving the system by increasing optical resolution, frame rate, feature generation/matching performance etc, would only improve upon this result.

5.2 Recommendations for Future Work

The conclusions in this chapter validated many of the original assertions and design decisions, but the limitations of the particular test apparatus left some conclusions incomplete, and did not allow the exploration of the SLAAMR algorithm's full potential. This section makes recommendations for future work that will improve upon the test apparatus and expand the envelope for its use. The following recommendations should be followed when attempting to use a SLAM-based system for INS aiding.

5.2.1 Optimize Feature Generation/Matching Algorithm for Feature Persistence. It was shown that dwell time and landmark tracking angle drive the GDOP and precision of the self-surveyed maps used by the SLAAMR algorithm. These are ultimately driven by the ability to find the same landmark feature/landmark in successive frames. Unfortunately, this is the most difficult and least understood portion of the SLAAMR algorithm, leaving ample room for improvement. Many of the techniques and added aiding used to overcome poor image processing may not be needed. Other studies have shown the benefit of good feature persistence, but do not employ a robust algorithm that would work in realistic environments.

5.2.2 Increase Camera Frame Rate. It was shown that increasing camera frame rate gives more measurements to the extended Kalman filter and increases performance. Another benefit lies in the feature matching problem. Faster frame rates would reduce the change in the scene from frame to frame (as less time has gone by). Features would also be expected to change less, and thus be easier to match. The frame rate used in test

was driven by hardware and computational limitations, and found to be inadequate for the higher *V over h* data points.

5.2.3 Explore the Full V over h Spectrum. The angular rate of features in the scene (driven by *V over h*) proved to be the most challenging problem for the system under test. Thus, only a portion of the spectrum was confidently evaluated (0.3-0.07 sec^{-1}). The recommendations above would, in theory, allow this envelope to be extended above 0.3 sec^{-1}. Some attention should also be paid to the very low *V over h* ratios (below 0.07 sec^{-1}) at very high altitudes (10,000-60,000 ft), as many unmanned vehicles may be expected to fly there.

5.2.4 Re-Evaluate the Zero A Priori Information Navigation Performance. Other studies [4][5][6][11] showed the promise of the zero a priori information navigation, but were not plagued by the image processing deficiencies of the system under test. A system improved by the above recommendations should perform more closely to predictions.

5.3 Summary

In summary, this thesis rigorously explored the nature of bearings-only optical measurements and their aiding benefit to an INS using a SLAM-based system. The theories were generally validated, but deficiencies in the system under test leave room for further exploration. The practical limitations of optics, image processing and computational power greatly affect the performance of a SLAM system, but can be overcome with the added measurement of altitude and heading. All in all, the SLAAMR

algorithm as described in this report provides passive, stealthy, and jamming-resistant navigation-grade INS performance in robust environments.

Bibliography

1. Alison A. Proctor, and Eric N. Johnson. *Vision-only Approach and Landing*. AIAA Guidance, Navigation, and Control Conference and Exhibit 15 - 18 August 2005, AIAA 2005-5871, Georgia Institute of Technology.

2. Allen D. Wu, Eric N. Johnson, and Alison A. Proctor. *Vision-Aided Inertial Navigation for Flight Control*. AIAA Guidance, Navigation, and Control Conference and Exhibit 15 - 18 August 2005, AIAA 2005-5998, Georgia Institute of Technology.

3. Brown, Alison K. and Lu, Yan, NAVSYS Corporation. *Performance Test Results of an Integrated GPS/MEMS Inertial Navigation Package*. Proceedings of ION GNSS 2004, Long Beach, California, September, 2004.

4. Brown, Alison and Silva, Randy, NAVSYS Corporation. *Video-aided GPS/INS Positioning and Attitude Determination*. ION GPS '99, Nashville, TN, September, 1999.

5. George, Sukkarieh. *Inertial Navigation Aided by Monocular Camera Observations of Unknown Features*. 2007 IEEE International Conference on Robotics and Automation Roma, Italy, 10-14 April 2007.

6. Giebner, Michael G. *Tightly-Coupled Image-Aided Inertial Navigation System via a Kalman Filter*. MS Thesis, Air Force Institute of Technology, Wright Patterson AFB, March, 2003.

7. Godha, Saurabh. *Performance Evaluation of Low Cost MEMS-Based IMU Integrated With GPS for Land Vehicle Navigation Application*. MS Thesis, UCGE Reports Number 20239, February, 2006.

8. Hoshizaki, Andrisani, Braun, Mulyana and Bethel, Purdue University. *Performance of Integrated Electro-Optical Navigation Systems.* Journal of the Institute of Navigation, Vol. 51, No. 2, Summer 2004.

9. Johnson, E. N. and Schrage, D. P. *The Georgia Tech Unmanned Aerial Research Vehicle: GTMax*. AIAA Guidance, Navigation and Control Conference, No. AIAA-2003-5741, Austin, TX, August 2003.

10. Jonghyuk Kim, and Salah Sukkarieh. *Autonomous Airborne Navigation in Unknown Terrain Environments*. IEEE Transactions on Aerospace and Electronic Systems, Vol. 40, NO. 3, July 2004.

11. Kim and Sukkarieh. *6DoF SLAM aided GNSS/INS Navigation in GNSS Denied and Unknown Environments*. Journal of Global Positioning Systems, Vol. 4, No. 1-2: 120-128, July 2005.

12. Lowe, David G., Computer Science Department, University of British Columbia. *Object Recognition from Scale-Invariant Features*. Proceedings of the International Conference on Computer Vision ICCV, Corfu.

13. Ma, Y., Soatto, S., Kosecka, J., Sastry, S. *An Invitation to 3-D Vision*. Springer, 2004.

14. Maybeck, Peter S. *Stochastic Models, Estimation, and Control Volume 1, 141-1*. Mathematics in Science and Engineering. Navtech Book & Software Store, republished with permission of author, 1994.

15. Maybeck, Peter S. *Stochastic Models, Estimation, and Control Volume 2, 141-1*. Mathematics in Science and Engineering. Navtech Book & Software Store, republished with permission of author, 1994.

16. Maybeck, Peter S. *Stochastic Models, Estimation, and Control Volume 3, 141-1*. Mathematics in Science and Engineering. Navtech Book & Software Store, republished with permission of author, 1994.

17. Nielsen, Bradley, Itsara, Grac, Viarengo. *Limited Evaluation of the SLAAMR Inertial Optical Sensor Fusion Algorithm.* Final Technical Memorandum, USAF Test Pilot School, December 2007.

18. Pachter, Meir. *Course Handouts, EENG 635, Inertial Navigation Subsystems*. Spring, 2006.

19. Pachter, Meir. *Course Handouts, EENG 735, Inertial Navigation Subsystems. S*ummer, 2006.

20. Pachter, Porter and Polat. *INS Aiding Using Bearings-Only Measurements of and Unknown Ground Object*. Paper, Air Force Institute of Technology, Wright Patterson AFB, September, 2005.

21. Raquet, John. *Course Handouts, EENG 533, Navigation Using the GPS.* Air Force Institute of Technology, Wright Patterson AFB, Spring, 2006.

22. Raquet, John F and Giebner, Michael. *Navigation Using Optical Measurements of Objects at Unknown Locations*. Proceedings of ION 59th Annual Meeting/CIGTF 22nd Guidance Test Symposium, 23-25 June 2003, Albuquerque, NM.

23. Shane Pinder, David Boid, Dan Sullivan, and Alison Brown, NAVSYS Corporation. *Navigation and Electro/Optic Sensor Integration Technology*. 58th Annual ION Meeting, Albuquerque, NM, June 2002.

24. Srikanth Saripalli, James F. Montgomery and Gaurav S. Sukhatm. *Vision-based Autonomous Landing of an Unmanned Aerial Vehicle*. Proceedings Of IEEE International Conference on Robotics and Automation pp 2799-2804, May 2002, Washington D.C., USA.

25. Titterton, David and Weston, John. *Strapdown Inertial Navigation Technology, 2nd Edition.* IEE Radar, Sonar and Navigation series 17, 2004.

26. Vasquez, J. *Course Handouts, EENG 510, Linear Systems.* Fall 2005.

27. Veth, Michael J. *Optical and Inertial Sensor Fusion.* PhD Dissertation. Air Force Institute of Technology, Wright Patterson AFB, September, 2006.

28. Veth M., Student Member, IEEE, John Raquet, Member, IEEE, Meir Pachter, Fellow, IEEE, Air Force Institute of Technology. *Stochastic Constraints for Fast Image Correspondence Search with Uncertain Terrain Model*. Paper, Air Force Institute of Technology, Wright Patterson AFB, 2007.

29. Veth, Mike and Raquet, John, Air Force Institute of Technology. *Two-Dimensional Stochastic Projections for Tight Integration of Optical and Inertial Sensors for Navigation*. Paper, Air Force Institute of Technology, Wright Patterson AFB, 2006.

30. Vidal-Calleja, Bryson, Sukkarieh, Sanfeliu and Andrade-Cetto. *On the Observability of Bearing-only SLAM*. Robotics and Automation, 2007 IEEE International Conference on 10-14 April 2007 Page(s):4114 – 4119.

31. *Boeing (McDonnell Douglas) JDAM (GBU-29/B, GBU-30/B, GBU-31/B, GBU-32/B, GBU-34/B, GBU-35/B, GBU-38/B)*. Informational webpage, http://www.designation-systems.net/dusrm/app5/jdam.html

32. *Embedded GPS/Inertial Navigation System (EGI).* Informational webpage, http://www.globalsecurity.org/space/systems/egi.htm

33. *PL-A741 Machine Vision Camera System Guide*. Informational webpage. http://ftp.elvitec.fr/PixeLINK/MANUELS/PL-A741%20Manuel.pdf

34. *The MIDG II INS/GPS.* Informational webpage. http://microboticsinc.com/ins_gps.html

35. *Directory of U.S. Military Rockets and Missiles.* Informational webpage. http://www.designation-systems.net/dusrm/app5/

36. Zhengyou Zhang, Senior Member, IEEE. *A Flexible New Technique For Camera Calibration.* IEEE Transactions on Pattern Analysis and Machine Intelligence, VOL. 22, NO. 11, November 2000.

Vita

Captain Michael B. Nielsen attended the University of Colorado at Boulder, majoring in electrical and computer engineering. While there, he was enrolled in the Air Force Reserve Officer Training Corps (AFROTC). Captain Nielsen graduated with distinction in 1999, and received his commission as a Second Lieutenant in the United States Air Force.

After graduation, Captain Nielsen attended the Euro NATO Joint Jet Pilot Training (ENJJPT) program, finishing a Distinguished Graduate in 2001. He was then assigned to the 333rd FTS for F-15E conversion training before being assigned to the 494th FS at RAF Lakenheath, England. There, Captain Nielsen served as a front line combat pilot in both Operation Iraqi Freedom and Operation Enduring Freedom, earning two Air Medals.

Captain Nielsen was selected for the joint Air Force Institute of Technology/United States Air Force Test Pilot School (AFIT/TPS) program in 2005. He majored in electrical engineering before leaving for Edwards, AFB to attend TPS. After graduation from TPS, Captain Nielsen was assigned to the 416 FLTS as a test pilot in the F-16.

The rest of the details of Capt Nielsen's life are wholly uninteresting and irrelevant to the discussion herein.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 074-0188

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From – To)*
27-03-2008	Master's Thesis	August 2005 – March 2008

4. TITLE AND SUBTITLE

DEVELOPMENT AND FLIGHT TEST OF A ROBUST OPTICAL-INERTIAL NAVIGATION SYSTEM USING LOW-COST SENSORS

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Nielsen, Michael B., Captain, USAF

5d. PROJECT NUMBER
JON 07-248

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAMES(S) AND ADDRESS(S)
Air Force Institute of Technology
Graduate School of Engineering and Management (AFIT/EN)
2950 Hobson Way, Building 640
WPAFB OH 45433-8865

8. PERFORMING ORGANIZATION REPORT NUMBER

AFIT/GE/ENG/08-19

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Air Force Research Laboratory Munitions Directorate
ATTN: AFRL/RWGI Timothy Klautsutis (Timothy.Klausutis@eglin.af.mil)
101 W. Eglin Blvd. (Phone: DSN 875-0887)
Eglin AFB, FL 32542

10. SPONSOR/MONITOR'S ACRONYM(S)
AFRL RWGI

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

This report develops and tests a precision navigation algorithm fusing optical and inertial measurements. This algorithm provides an alternative to the Global Positioning System (GPS) as a precision navigation source, enabling navigation in GPS denied environments, using low-cost sensors and equipment.

A rigorous study of the fundamental nature of optical/inertial navigation is accomplished through the examination of the observability grammian of the underlying measurement equations. Through this analysis, a set of principles of design are proposed guiding the development of a navigation algorithm. An algorithm titled the Simultaneous Location Aiding And Mapping Recursively (SLAAMR) is thus described incorporating these principles of design, an extended Kalman Filter (EKF) and a Simultaneous Location And Mapping (SLAM) process. The SLAAMR algorithm is designed to provide robust navigation performance in realistic, full scale environments at a low cost.

The principles of design and the SLAAMR algorithm are tested and evaluated using data collected at the United States Air Force Test Pilot School (USAF TPS). A full-scale aircraft flying operationally representative parameters and profiles was used to collect the data, and was correlated with highly precise Time Space Position Information (TSPI) truth data for validation and evaluation purposes.

The flight test data supports the principles of design and highlights the challenges faced by any optically-based navigation system. The resultant performance of the SLAAMR algorithm provides a robust, practical navigation solution to Air Force aircraft.

15. SUBJECT TERMS

Robust Low Cost Monocular Optical Inertial Navigation, Flight Test, Observability Grammian, Aircraft, Landing, INS, IMU, Camera, Image Processing, Feature Landmark Tracking, extended Kalman Filter, SLAM, SLAAMR.

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			John Raquet, PhD
U	U	U	UU	295	19b. TELEPHONE NUMBER *(Include area code)* (937) 255-3636, ext 4580 (john.raquet@afit.edu)

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std. Z39-18