The Thesis Committee for Daniel Phillip Shepard
certifies that this is the approved version of the following thesis:

Fusion of Carrier-Phase Differential GPS, Bundle-Adjustment-Based Visual SLAM, and Inertial Navigation for Precisely and Globally-Registered Augmented Reality

APPROVED BY

SUPERVISING COMMITTEE:

Todd E. Humphreys, Supervisor

Glenn Lightsey

Fusion of Carrier-Phase Differential GPS, Bundle-Adjustment-Based Visual SLAM, and Inertial Navigation for Precisely and Globally-Registered Augmented Reality

by

Daniel Phillip Shepard, BSAsE

THESIS

Presented to the Faculty of the Graduate School of
The University of Texas at Austin
in Partial Fulfillment
of the Requirements
for the Degree of

MASTER OF SCIENCE IN ENGINEERING

THE UNIVERSITY OF TEXAS AT AUSTIN
May 2013

Acknowledgments

I would like to thank Dr. Todd Humphreys for his support and insightful conversations throughout the course of writing this thesis. I could not have asked for a better adviser. I would also like to thank the other members of the Radionavigation Laboratory, Ken Pesyna, Jahshan Bhatti, Reese Shetrone , Andrew Kerns, Zaher Kassas, and Kyle Wesson, who have all helped me out either through discussions, during development, or during testing of the work that went into this thesis. Finally, I would like to thank my family for their constant encouragement.

Fusion of Carrier-Phase Differential GPS, Bundle-Adjustment-Based Visual SLAM, and Inertial Navigation for Precisely and Globally-Registered Augmented Reality

Daniel Phillip Shepard, MSE
The University of Texas at Austin, 2013

Supervisor: Todd E. Humphreys

Methodologies are proposed for combining carrier-phase differential GPS (CDGPS), visual simultaneous localization and mapping (SLAM), and inertial measurements to obtain precise and globally-referenced position and attitude estimates of a rigid structure connecting a GPS receiver, a camera, and an inertial measurement unit (IMU). As part of developing these methodologies, observability of globally-referenced attitude based solely on GPS-based position estimates and visual feature measurements is proven. Determination of attitude in this manner eliminates the need for attitude estimates based on magnetometer and accelerometer measurements, which are notoriously susceptible to magnetic disturbances. This combination of navigation techniques, if coupled properly, is capable of attaining centimeter-level or better absolute positioning and degree-level or better absolute attitude accuracies in any space, both indoors and out. Such a navigation system is ideally suited for application to augmented reality (AR), which often employs a GPS receiver, a camera, and an IMU, and would result in tight registration of virtual elements to the real world. A prototype AR system is presented that represents a first step towards coupling CDGPS, visual SLAM, and inertial navigation. While this prototype AR system does not couple CDGPS and visual SLAM tightly enough to obtain some of the benefit of the proposed methodologies, the system is capable of demonstrating an upper bound on the precision that such a combination of navigation techniques could attain. Test results for the prototype AR system are presented for a dynamic scenario that demonstrate sub-centimeter-level positioning precision and sub-degree-level attitude precision. This level of precision would enable convincing augmented visuals.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Augmented reality (AR) is a concept closely related to virtual reality (VR), but has a fundamentally different goal. Instead of replacing the real world with a virtual one like VR does, AR seeks to produce a blended version of the real world and context-relevant virtual elements that enhance or augment the user's experience in some way, typically through visuals. The relation of AR to VR is best explained by imagining a continuum of perception with the real world on one end and VR on the other. On this continuum, AR would be placed in between the real world and VR with the exact placement depending on the goal of the particular application of AR.

AR has been a perennial disappointment since the term was first coined 23 years ago by Tom Caudell. Wellner et al. [5] in 1993 lamented that "for the most part our computing takes place sitting in front of, and staring at, a single glowing screen attached to an array of buttons and a mouse." As the ultimate promise of AR, he imagined a world where both entirely virtual objects and real objects imbued with virtual properties could be used to bring the physical world and computing together. Instead of viewing information on a two-dimensional computer screen, the three-dimensional physical world becomes a canvas on which virtual information can be displayed or edited either individually or collaboratively. Twenty years have passed since Wellner's article and little has changed. There have been technological advances in AR, but, with all the promise of AR, it simply has not gained much traction in the commercial world.

The operative question is then what has prevented AR from reaching Wellner's vision. The answer is that creating augmented visuals that provide a convincing illusion of realism is extremely difficult. Thus, AR has either suffered from poor alignment of the virtual elements and the real world, resulting in an unconvincing illusion, or has been limited in application to avoid this difficulty.

Errors in the alignment of virtual objects or information with their desired real world position and orientation, or pose, are typically referred to as registration errors. Registration errors are a direct result of the estimation error of the user's position and orientation relative to the virtual element. These registration errors have been the primary limiting factor in the suitability of AR for various applications [6]. If registration errors are too large, then it becomes difficult or even impossible to interact with the virtual objects because the object may not appear stationary as the user approaches. This is because registration errors become more prominent in the user's view of the object as the user gets closer to the virtual object due to user positioning errors.

Many current AR applications leverage the fact that user positioning errors have little impact on registration errors when virtual objects are far away and constrain themselves to only visualizing objects at a distance. The recently announced Google Glass [7] falls into this category. Some of these applications, including Google Glass, are described in Sec. 1.1. While there is utility to these applications, they seem disappointing when compared to Wellner's vision of a fully immersive AR experience.

Techniques capable of creating convincing augmented visuals with small registration errors have been created using relative navigation to visual cues in the environment. However, these techniques are not generally applicable. Relative navigation alone does not provide any global reference, which is necessary for many applications and convenient for others.

To reach the ultimate promise of AR envisioned by Wellner, we imagine an ideal AR system that is

Accurate: The ideal AR system should provide absolute camera pose with centimeter-level or better positioning accuracy and sub-degree-level attitude accuracy. For a positioning error of 1 cm and an attitude error of half a degree, a virtual object 1 m in front of the camera would have at most a registration error of approximately 1.9 cm in position.

Available: The ideal AR system should be capable of providing absolute camera pose at the above accuracy in any space, both indoors and out.

Inexpensive: The ideal AR system should be priced in a reasonable range for a typical consumer.

Easy to Use: The ideal AR system should be easy for users to either hold up in front of them or wear on their head. The augmented view should also be updated in real-time with no latency by propagating the best estimate of the camera pose forward in time through a dynamics model.

The desired positioning accuracy is difficult to achieve in a global reference frame, but can be accomplished with carrier-phase differential GPS (CDGPS). CDGPS, commonly referred to as real-time-kinematics (RTK) for operation in real-time with motion, is a technique in which the difference between the carrier-phase observables from two GPS receivers are used to obtain the relative position of the two antennas. Under normal conditions, this technique results in centimeter-level or better accuracy of the relative position vector. Therefore, if the location of one of the antennas, the reference antenna, is known accurately from a survey of the location, then the absolute coordinates of the other antenna, the mobile antenna, can be determined to centimeter-level or better accuracy.

Currently, the price of commercially available CDGPS-capable receivers is out of reach for the typical consumer. However, the price could easily be reduced by making concessions in regards to signal diversity. CDGPS-capable receivers currently on the market are designed primarily for surveyors that desire instant, high-accuracy position fixes, even in urban canyons. This requires the use of multiple satellite constellations and multiple signal frequencies. Each additional satellite constellation and signal frequency adds significant cost to the receiver. On the other hand, inexpensive, single-frequency GPS receivers are on the market that produce the carrier-phase and pseudorange observables required to obtain CDGPS accuracy.

The concession of reducing signal diversity to maintain price, however, exacerbates problems with GPS availability. GPS reception is too weak for indoor navigation and is difficult in urban canyons. Multiple constellations could help with urban canyons, but indoor navigation with GPS alone is a difficult problem.

One well published solution to address GPS availability issues and provide attitude estimates is to couple GPS-based positioning with an inertial navigation system (INS). The sensors for an INS typically consist of a three-axis accelerometer, a three-axis gyro, a magnetometer, and possibly a thermometer (for temperature calibration of the sensors). For the remainder of this thesis, the term inertial measurement unit (IMU) will be used to collectively refer to the sensors comprising an INS, as listed above. However, a coupled CDGPS and INS navigation system provides poor attitude estimates during dynamics and near magnetic disturbances. Additionally, the position solution of a coupled CDGPS and INS navigation system drifts quickly during periods of GPS unavailability for all but the highest-quality IMUs, which are large and expensive.

A far superior option, which is presented in this thesis, is to couple CDGPS with monocular visual simultaneous localization and mapping (SLAM). Visual SLAM is ideally situated as a complementary navigation technique to CDGPS-based navigation. This combination of navigation techniques is special in that neither one acting alone can observe globally-referenced attitude, but their combination allows globally-referenced attitude to be recovered, which is proven in subsequent chapters. Visual SLAM alone provides high-accuracy relative pose in areas rich with nearby visually recognizable features. These nearby feature rich environments include precisely the environments where GPS availability is poor or non-existent. During periods of GPS availability, CDGPS can provide the reference to a global coordinate system that visual SLAM lacks. During periods of GPS unavailability, visual SLAM provides pose estimates that drift much more slowly, relative to absolute coordinates, than all but the highest-quality IMUs. An INS with an inexpensive IMU could be combined with this solution for additional robustness, particularly during periods of GPS unavailability to further reduce the drift of the pose estimates. This fusion of navigation techniques has the potential to satisfy the ultimate promise of AR.

One example of an application that would benefit from the AR system described above is construction. Currently, construction workers must carefully compare building plans with measurements on site to determine where to place beams and other structural elements, among other tasks. Construction could be expedited with the ability to visualize the structure of a building in its exact future location while building the structure. In particular, Shin identified 8 of 17 construction tasks in [8] that could be performed more efficiently by employing AR technologies.

Another potential application of this AR system is utility work. Utility workers need to identify existing underground structure before digging to avoid damaging existing infrastructure and prevent accidents that may cause injury. AR would enable these workers to “see” current infrastructure and easily avoid it without having to interpret schematics and relate that to where they are trying to dig.

There are many other interesting consumer applications in areas like gaming, social media, and tourism that could be enabled by a low-cost, general purpose AR platform providing robust, high-accuracy absolute pose of the camera. An ideal AR system would be usable for all these applications and could operate in any space, both indoors and out. Much like a smart-phone, the AR system could provide an application programming interface (API) that other application specific software could use to request pose information and push augmented visuals to the screen.

1.1 Background

In relation to the lofty goals of an ideal AR system laid out above, current consumer AR technology fails to impress. Nevertheless, some isolated applications of AR for which the full realization of the ideal AR system is unnecessary have been successful. These applications typically rely on visual cues or pattern recognition for relative navigation, but there are some applications that leverage absolute pose which do not have as stringent accuracy requirements as those envisioned for the ideal AR system. The following are some of these applications:

Sports Broadcasts: Sports broadcasts have used limited forms of AR for years to overlay information on the video feed to aid viewers. One example of this is the line-of-scrimmage and first-down lines typically drawn on American Football broadcasts. This technology uses a combination of visual cues from the footage itself and the known location of the video cameras [9]. This technology can also be seen in broadcasts of the Olympic Games for several sports including swimming and many track and field events. In this case, the lines drawn on the screen typically represent record paces or markers for previous athletes' performances.

Lego Models: To market their products, Lego employs AR technology at their kiosks which displays the fully constructed Lego model on top of the product package when held in front of a smart-phone camera. This technique uses visual tags on the product package to position and orient the model on top of the box [10].

Word Lens: Tourists to foreign countries often have trouble finding their way around because the signs are in foreign languages. Word Lens is an AR application which translates text on signs viewed through a smart-phone camera as illustrated in Fig. 1.1 [1]. This application uses text recognition software to identify portions of the video feed with text and then places the translated text on top of the original text with the same color background.

Wikitude: Wikitude is another smart-phone application which displays information about nearby points of interest, such as restaurants and landmarks, in text bubbles above their actual location as the user looks around while holding up their smart-phone [11]. This application leverages coarse pose estimates provided by GPS and an IMU.

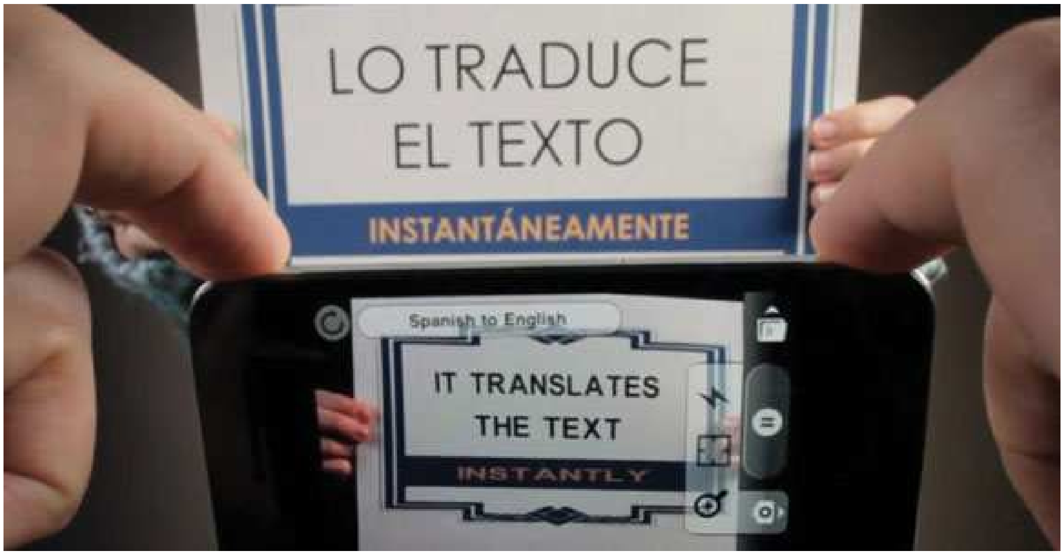

Figure 1.1: A smart-phone running the Word Lens application that is translating a sign from Spanish to English [1].

StarWalk: StarWalk is an application for smart-phones which allows users to point their smart-phones toward the sky and display constellations in that portion of the sky [2]. Like Wikitude, StarWalk utilizes coarse pose estimates provided by GPS and an IMU. However, StarWalk does not overlay the constellations on video from the phone. The display is entirely virtual, but reflects the user's actual pose.

Layar: Layar began as a smart-phone application that used visual recognition to overlay videos and website links onto magazine articles and advertisements [12]. The company, also called Layar, later created a software development kit that allows others to create their own AR applications based on either visual recognition, pose estimates provided by the smart-phone, or both. Like this thesis, Layar imagines a world where AR becomes a part of our everyday lives and seeks to bring the world closer to that goal.

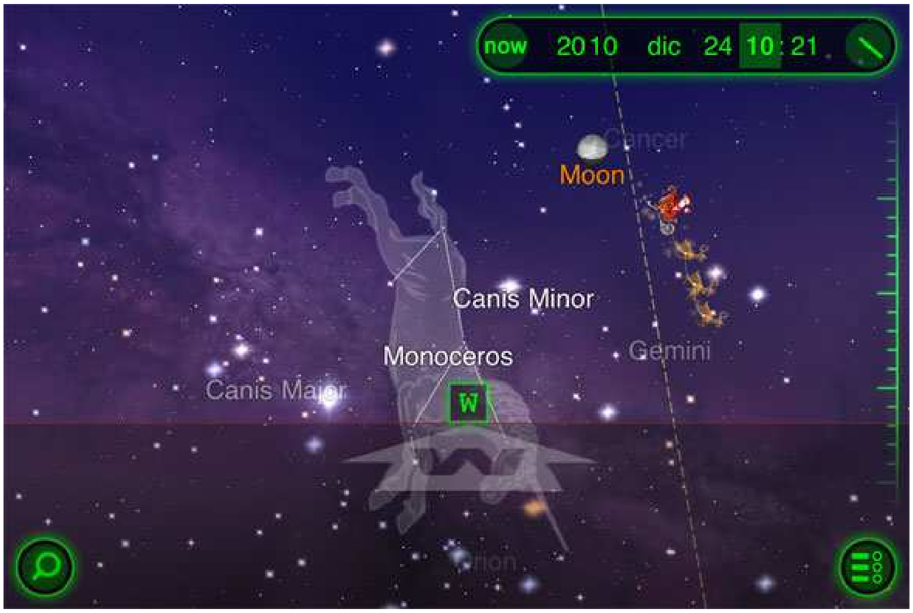

Figure 1.2: The Star Walk application is being used to highlight constellations in the night sky [2].

Google Glass: Google recently introduced a product called Glass which is a wearable AR platform that looks like a pair of glasses with no lenses and a small display above the right eye. This is easily the most ambitious consumer AR platform to date. However, Glass makes no attempt toward improving registration accuracy over existing consumer AR. Glass is essentially just a smart-phone that is worn on the face with some additional hand gestures for ease of use. Like a smart-phone, Glass has a variety of useful applications that are capable of tasks such as giving directions, sending messages, taking photos or video, making calls, and providing a variety of other information on request [7].

1.2 Prior Work

1.2.1 Augmented Reality

Prior work in AR can be divided into two primary categories, fiduciary-marker-based and non-fiduciary-marker-based. Work in each of these categories is discussed separately below. This discussion is restricted to those techniques which provide or have the potential to provide absolute pose.

1.2.1.1 Fiduciary-Marker-Based Augmented Reality

Fiduciary-marker-based AR relies on identification of visual cues or markers that can be correlated with a globally-referenced database and act as anchors for relative navigation. This requires the environment in which the AR system will operate to either be prepared, by placing and surveying fiduciary markers, or surveying for native features which are visually distinguishable ahead of time.

One such fiduciary AR technique by Huang et al. uses monocular visual SLAM to navigate indoors by matching doorways and other room-identifying-features to an online database of floor plans [13]. The appropriate floor plan is found using the rough location provided by an iPhone's or iPad's hybrid navigation algorithm, which is based on GPS, cellular phone signals, and Wi-Fi signals. The attitude is based on the iPhone's or iPad's IMU. This information was used to guide the user to locations within the building. The positioning of this technique was reported as accurate to meter-level, which would result in large registration errors for a virtual object within a meter of the user.

Another way of providing navigation for an AR system is to place uniquely identifiable markers at surveyed locations, like on the walls of buildings or on the ground. AR systems could download the locations of these markers from an online database as they identify the markers in their view and position themselves relative to the markers. This is similar to what is done with survey markers, which are often built into sidewalks and used as a starting point for surveyors with laser ranging equipment. An example of this technique used in a visual SLAM framework is given in [14] by Zachariah et al. This particular implementation uses a set of visual tags on walls in a hallway seen by a monocular camera and an IMU. Decimeter-level positioning accuracy was obtained in this example, which would still result in large registration errors for a virtual object within a meter of the user. This method also does not scale well as it would require a dense network of markers to be placed everywhere an AR system would be operated.

A final method takes the concept of fiduciary markers to its extreme limit and represents the current state of the art in fiduciary-marker-based AR. This technique is based on Microsoft's PhotoSynth which was pioneered by Snavely et al. in [15]. PhotoSynth takes a crowd-sourced database of photos of a location and determines the calibration and pose of the camera for each picture and the location of identified features common to the photos. PhototSynth also allows for smooth interpolation between views to give a full 6 degree-of-freedom (DOF) explorable model of the scene. This feature database could be leveraged for AR by applying visual SLAM and feature matching with the database after narrowing the search space with a coarse position estimate. In a TED talk by Arcas of Bing Maps [16] in 2010, the power of this technique for AR was demonstrated through a live video of Arcas' colleagues from a remote location that was integrated into Bing Maps as a floating frame at the exact pose of the real world video camera.

While the PhotoSynth approach seems to satisfy the accuracy requirements of an ideal AR system, there are several problems to universal availability. First, this technique requires that the world be mapped with pictures taken from enough angles for PhotoSynth to work. This could be crowd-sourced for many locations that are public and well trafficked, but other areas would have to be explored specifically for this purpose. Google and Microsoft both have teams using car and backpack mounted systems to provide street views for their corresponding map programs which could be leveraged for this purpose. However, the area covered by these teams is insignificant when it comes to mapping the whole world. Second, the world would have to be mapped over again as the environment changes. This requires a significant amount of management of an enormously large database. Third, applications that operate in changing environments, such as construction, could not use this technique. Finally, private spaces will require those who use the space to take these images themselves. For people to use this technique in their homes, they would need to walk around their homes and take pictures of every room from a number of different angles and locations. In addition to being a hassle for users, this could also create privacy issues if these images had to be incorporated into a public database to be usable with AR applications. Communications bandwidth would also be a severe limitation to the proliferation of AR using this technique.

1.2.1.2 Non-Fiduciary-Marker-Based Augmented Reality

Non-fiduciary-marker-based AR providing absolute pose primarily, if not entirely, consists of GPS-based solutions. Most of these systems couple some version of GPS positioning with an IMU for attitude. Variants of GPS positioning that have been used are

1. pseudorange-based GPS, which, for civil users, provides meter-level positioning accuracy and is referred to as the standard positioning service (SPS).

2. differential GPS (DGPS), which provides relative positioning to a reference station at decimeter-level accuracy.

3. carrier-phase differential GPS (CDGPS), which provides relative positioning to a reference station at centimeter-level accuracy or better.

One of the first GPS-based AR systems was designed to aid tourists in exploring urban environments. This AR system was developed in 1997 by Feiner et al. at Columbia University [3]. Feiner's AR system, shown in Fig. 1.3, is composed of a backpack with a computer and GPS receiver, a pair of goggles for the display with a built-in IMU, and a hand-held pad for interfacing with the system. The operation of this system is similar to Wikitude, mentioned in Sec. 1.1, in that it overlays information about points of interest on their corresponding location and aids the user in navigating to these locations. In fact, the reported pose accuracy of this device is comparable to that of Wikitude even though this system uses DGPS. The fact that the GPS antenna is not rigidly attached to the IMU and display also severely limits the potential accuracy of this AR system configuration even if the positioning accuracy of the GPS receiver was improved.

An AR system similar to the Columbia system was created and tested by Behzadan et al. [17, 18] at the University of Michigan for visualizing construction work-flow. Initially the AR system only used SPS GPS with a gyroscopes-only attitude solution, but was later upgraded with DGPS and a full INS.

Roberts et al. at the University of Nottingham built a hand-held AR system that looks like a pair of binoculars which allows utility workers to visualize subsurface infrastructure [4, 19]. An example of the visuals obtained from this system is shown in Fig. 1.4. This AR system used an uncoupled CDGPS and IMU solution for its pose

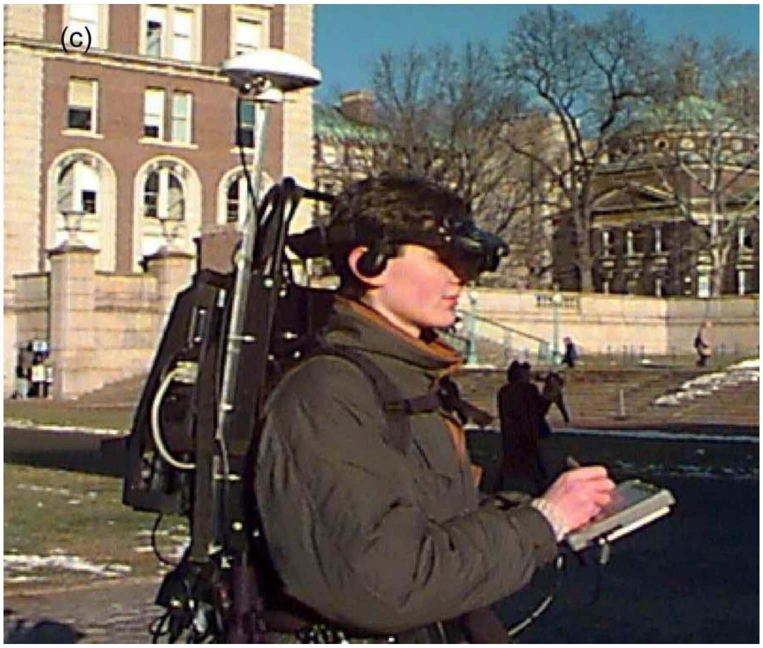

Figure 1.3: An augmented reality system developed by researchers at Columbia University for providing users with information about their surroundings [3].

estimate. However, no quantitative analysis of the system's accuracy was presented. This AR system restricts the user to applications with an open sky view, since it cannot produce position estimates in the absence of GPS. In a dynamic scenario, the CDGPS position solution would also suffer from the unknown user dynamics. The IMU could easily alleviate this issue if it were coupled to the CDGPS solution.

Schall et al. also constructed a hand-held AR device for visualizing subsurface infrastructure at Graz University of Technology [20]. Although their initial prototype only used SPS GPS and an IMU, much effort was spent in designing software to provide convincing visualizations of the subsurface infrastructure and on the ergonomics of the device. Later papers report an updated navigation filter and AR system that loosely couples CDGPS, an IMU, and a variant of visual SLAM for drift-free attitude tracking [21, 22]. This system comes the closest to that presented in this thesis,

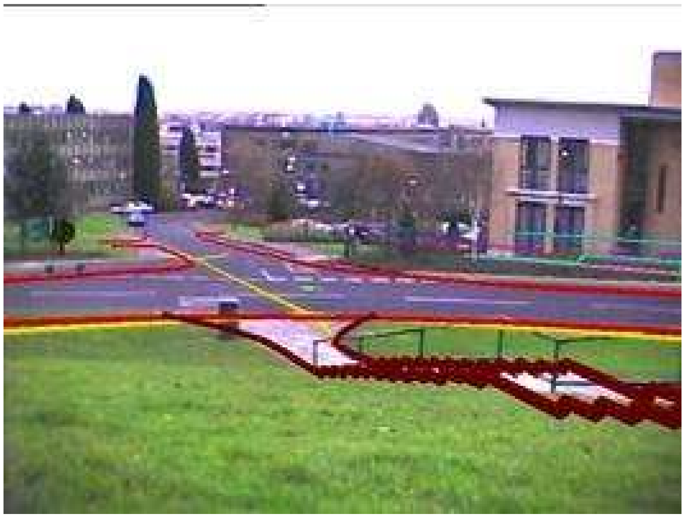

Figure 1.4: An image produced by an AR system developed by researchers at the University of Nottingham showing subsurface utilities overlaid onto the real world image in the background [4].

but does not fully couple CDGPS and visual SLAM. Further discussion on this AR system's navigation filter is given in the following section.

1.2.2 Vision-Aided Navigation

Vision-aided navigation couples some form of visual navigation with other navigation techniques to improve the navigation system's performance. The vast majority of prior work in vision-aided navigation has only coupled visual SLAM and an INS. This allows for resolution of the inherent scale-factor ambiguity of the map created by visual SLAM to recover true metric distances. This approach has been broadly explored in both visual SLAM methodologies, filter-based and bundle-adjustment-based. Examples of this approach for filter-based visual SLAM and bundle-adjustment-based visual SLAM are given in [23–26] and [27–29] respectively. Several papers even specifically mention coupled visual SLAM and INS as an alternative to GPS, instead of a complementary navigation technique [30, 31].

There has been some prior work on coupling visual navigation and GPS, but these techniques only coupled the two in some limited fashion. One example of this is a technique developed by Soloviev and Venable that used GPS carrier-phase measurements to aid in scale-factor resolution and state propagation in an extended Kalman filter (EKF) visual SLAM framework [32]. This technique was primarily targeted at GPS-challenged environments where only a few GPS satellites could be tracked. Another technique developed by Wang et al. only used optical flow to aid a coupled GPS and INS navigation solution for an unmanned aerial vehicle [33].

The closest navigation technique to a full coupling of GPS and visual SLAM was developed by Schall et al., as previously mentioned [21, 22]. An important distinction of Schall's filter from a fully-coupled GPS and visual SLAM approach is that Schall's filter only extracts attitude estimates from visual SLAM to smooth out the IMU attitude estimates. In fact, Schall's filter leaves attitude estimation and position estimation decoupled and does not use accelerometer measurements from the IMU for propagating position between GPS measurements. This approach limits the absolute attitude accuracy of the filter to that of the IMU, which is not true for a fully-coupled GPS, visual SLAM, and IMU approach. This filter is also sub-optimal in that it throws away positioning information that could be readily obtained from the visual SLAM algorithm, ignores accelerometer measurements, and ignores coupling between attitude and position.

1.3 Contributions

In contrast to other approaches that combine GPS and visual SLAM in a limited fashion, this thesis presents methods to fully fuse GPS and visual SLAM that would enable convincing absolute registration in any space, both indoors and out. One added benefit to this coupling is the recovery of absolute attitude without the use of an IMU. A sufficient condition for observability of the locations of visual features and the absolute pose of the camera without the use of an IMU is presented and proven. Several potential filter architectures are presented for combining GPS, visual SLAM, and an INS and the advantages of each are discussed. These filter architectures include an original filter-based visual SLAM method that is a modified version of the method presented by Mourikis et al. in [23].

A filter that combines CDGPS, bundle-adjustment-based visual SLAM, and an INS is presented which, while not optimal, is capable of demonstrating the potential of this combination of navigation techniques. A prototype AR system based on this filter is detailed and shown to obtain accuracy that would enable convincing absolute registration. With some modification to the prototype AR system so that visual SLAM is coupled tighter to the navigation system, this AR system could operate in any space, indoors and out. Further prototypes of the AR system could be miniaturized and reduced in cost with little effect on the accuracy of the system in order to approach the ideal AR system.

1.4 Organization

The remainder of the thesis is structured as follows

Chapter 2 gives an observability proof for combined GPS and visual SLAM and discusses potential approaches for optimal fusion of GPS, visual SLAM, and an INS in the sense of accuracy per computational cost.

Chapter 3 presents the measurement and dynamics models used in the navigation filter of the prototype AR system.

Chapter 4 defines the navigation filter of the prototype AR system.

Chapter 5 describes the hardware and software used for the prototype AR system.

Chapter 6 presents test results for the prototype AR system.

Chapter 7 gives conclusions for this thesis and describes areas for future work.

Appendices present some mathematical background in quaternion math, the square-root EKF (SREKF), and integer ambiguity resolution that is helpful for comprehension of the navigation filter presented in this thesis.

Chapter 2

Analysis of Coupled Visual SLAM and GPS

In recent years, vision-aided inertial navigation has received much attention as a method for resolving the scale-factor ambiguity inherent to monocular visual SLAM. With the scale-factor ambiguity resolved, high-accuracy relative navigation has been achieved. This method has widely been considered an alternative to GPS-based absolute pose techniques, which have problems navigating in urban canyons and indoors. Few researcher have coupled visual SLAM with GPS, and those who have only did so in a limited fashion.

This chapter investigates how these two complementary navigation techniques and inertial measurements can be coupled with the goal of obtaining highly accurate absolute pose in any area of operation, indoors and out. The chapter begins with an observability analysis which demonstrates that absolute pose can be recovered by combining visual SLAM and GPS alone. This combination of measurements is special in that neither one acting alone can observe absolute attitude, but their combination allows absolute attitude to be recovered. A discussion of estimation methods is then presented that details the unique aspects of the visual SLAM problem from an estimation standpoint. Estimation strategies are detailed and compared for the problems of stand-alone visual SLAM and coupled visual SLAM, GPS, and inertial sensors.

2.1 Observability Analysis

Consider a rigid body on which is rigidly mounted a calibrated camera. The body frame of the rigid body will be taken as the camera reference frame and denoted as $\mathcal{C}$. Its origin and x and y axes lie in the camera's image plane; its z axis points down the camera bore-sight. A reference point $\mathbf{x}^r = [x^r, y^r, z^r]^T$ is fixed on the rigid body. When expressed in the camera frame, $\mathbf{x}^r$ is written $\mathbf{x}^r_{\mathcal{C}} = [x^r_{\mathcal{C}}, y^r_{\mathcal{C}}, z^r_{\mathcal{C}}]^T$ and is constant. Consider a scene viewed by the camera that consists of a collection of M static point features in a local reference frame $\mathcal{L}$. The jth point feature has constant coordinates in frame $\mathcal{L}$: $\mathbf{x}^{p_j}_{\mathcal{L}} = [x^{p_j}_{\mathcal{L}}, y^{p_j}_{\mathcal{L}}, z^{p_j}_{\mathcal{L}}]$.

The camera moves about the static point features and captures N keyframes, which are images of the M point features taken from distinct views of the scene. A distinct view is defined as a view of the scene from a distinct location. Although not required by the definition, these distinct views may also have differing attitude so long as the M point features remain in view of the camera. Each keyframe has a corresponding reference frame $\mathcal{C}_i$, which is defined to be aligned with the camera frame at the instant the image was taken, and image frame $\mathcal{I}_i$, which is defined as the plane located 1 m in front of the camera lens and normal to the camera bore-sight. It is assumed that the M point features are present in each of the N keyframes and can be correctly and uniquely identified.

To determine the projection of the M point features onto the image frames of the N keyframes, the point features are first expressed in each $\mathcal{C}_i$. This operation is expressed as follows

$$\mathbf{x}^{p_j}_{\mathcal{C}_i} = R(\mathbf{q}^{\mathcal{L}}_{\mathcal{C}_i})(\mathbf{x}^{p_j}_{\mathcal{L}} - \mathbf{x}^{\mathcal{C}_i}_{\mathcal{L}}), \text{ for } i = 1, 2, \ldots, N \ \& \ j = 1, 2, \ldots, M \tag{2.1}$$

where $\mathbf{q}_{\mathcal{C}_i}^{\mathcal{L}}$ is the quaternion representation of the attitude of the camera for the ith keyframe relative to the $\mathcal{L}$ frame, $R(\cdot)$ is the rotation matrix corresponding to the argument, and $\mathbf{x}_{\mathcal{L}}^{\mathcal{C}_i}$ is the position of the origin of the camera (hereafter the camera position) for the ith keyframe expressed in the $\mathcal{L}$ frame. For any attitude representation, $(\cdot)_{\mathcal{B}}^{\mathcal{A}}$ represents a rotation from the $\mathcal{A}$ frame to the $\mathcal{B}$ frame. Details on quaternion representation of attitude and quaternion math are provided in Appendix A.

A camera projection function $p(\cdot)$ converts a vector expressed in the camera frame $\mathcal{C}_i$ into a two-dimensional projection of the vector onto the image frame $\mathcal{I}_i$ as

$$\mathbf{s}_{\mathcal{I}_i}^{p_j} = \begin{bmatrix} \alpha_{\mathcal{I}_i}^{p_j} \\ \beta_{\mathcal{I}_i}^{p_j} \end{bmatrix} = p(\mathbf{x}_{\mathcal{C}_i}^{p_j}), \text{ for } i = 1, 2, \ldots, N \text{ \& } j = 1, 2, \ldots, M \tag{2.2}$$

The set of these projected coordinates for each point feature and each keyframe constitute the measurements provided by a feature extraction algorithm operating on these keyframes.

Suppose that, in addition to these local measurements, measurements of the position of the reference point on the rigid body are provided in a global reference frame $\mathcal{G}$ at each keyframe, denoted as $\mathbf{x}_{\mathcal{G}}^{r_i}$. The position of the reference point in $\mathcal{G}$ is related to the pose of the camera through the equation

$$\mathbf{x}_{\mathcal{G}}^{r_i} = \mathbf{x}_{\mathcal{G}}^{\mathcal{C}_i} + R(\mathbf{q}_{\mathcal{G}}^{\mathcal{C}_i})\mathbf{x}_{\mathcal{C}}^{r}, \text{ for } i = 1, 2, \ldots, N \tag{2.3}$$

The local frame $\mathcal{L}$ is fixed with respect to $\mathcal{G}$ and is related to $\mathcal{G}$ by a similarity transform. A vector expressed in $\mathcal{G}$ can be expressed in $\mathcal{L}$ through the equation

$$\mathbf{x}_{\mathcal{L}} = \lambda R(\mathbf{q}_{\mathcal{L}}^{\mathcal{G}})(\mathbf{x}_{\mathcal{G}} + \mathbf{x}_{\mathcal{G}}^{\mathcal{L}}) \tag{2.4}$$

where $\mathbf{x}_{\mathcal{G}}^{\mathcal{L}}$, $\mathbf{q}_{\mathcal{L}}^{\mathcal{G}}$, and λ are the translation, rotation, and scale-factor that characterize the similarity transform from $\mathcal{G}$ to $\mathcal{L}$.

The globally-referenced structure from motion problem can be formulated as follows: Given the measurements $\mathbf{s}_{\mathcal{I}_i}^{p_j}$ and $\mathbf{x}_{\mathcal{G}}^{r_i}$ for $i = 1, 2, \ldots, N$ and $j = 1, 2, \ldots, M$, estimate the camera pose for each frame (parameterized by $\mathbf{x}_{\mathcal{L}}^{\mathcal{C}_i}$ and $\mathbf{q}_{\mathcal{L}}^{\mathcal{C}_i}$ for $i = 1, 2, \ldots, N$), the location of each point feature ($\mathbf{x}_{\mathcal{L}}^{p_j}$ for $j = 1, 2, \ldots, M$), and the similarity transform relating $\mathcal{G}$ and $\mathcal{L}$ (parameterized by $\mathbf{x}_{\mathcal{G}}^{\mathcal{L}}$, $\mathbf{q}_{\mathcal{L}}^{\mathcal{G}}$, and λ).

The goal of the following analysis is to define a set of sufficient conditions under which these quantities are observable. To start, the projection function from Eq. 2.2 is taken to be a perspective projection and weak local observability is tested. A proof of weak local observability only demonstrates that there exists a neighborhood around the true value inside which the solution is unique, but not necessarily a globally unique solution. Stronger observability results are then proven under the more restrictive assumption that the projection is orthographic.

2.1.1 Perspective Projection Analysis

A perspective projection, also known as a central projection, projects a view of a three-dimensional scene onto an image plane through rays connecting three-dimensional locations and a center of projection. This is the type of projection that results from a camera image. A perspective projection can be expressed mathematically, assuming a calibrated camera, as

$$p(\mathbf{x}_{\mathcal{C}}) = \frac{1}{z_{\mathcal{C}}} \begin{bmatrix} x_{\mathcal{C}} \\ y_{\mathcal{C}} \end{bmatrix} \tag{2.5}$$

To demonstrate weak local observability, the measurements from Eqs. 2.2 and 2.3 were linearized about the true values of the camera poses and the feature locations in $\mathcal{G}$. The resulting matrix was tested for full column rank under a series of scenarios. This test is a necessary and sufficient condition for weak local observability, which means the solution is unique within a small neighborhood about the true values of the quantities to be estimated but not necessarily globally unambiguous.

The weak local observability tests revealed that with as few as three keyframes of three point features the problem is fully locally observable provided the following conditions are satisfied:

1. The three feature points are not collinear.

2. The positions of the camera for each frame are not collinear.

3. The positions of the reference point for each frame are not collinear.

2.1.2 Orthographic Projection Analysis

An orthographic projection projects a view of a three-dimensional scene onto an image plane through rays parallel to the normal of the image plane. Although this projection does not describe how images are formed in a camera, this is a good approximation to a perspective projection in a small segment of the image, so long as the distance from the camera to the point features is much larger than the distance between the point features [34]. An orthographic projection can be expressed mathematically as

$$p(\mathbf{x}_{\mathcal{C}}) = \begin{bmatrix} x_{\mathcal{C}} \\ y_{\mathcal{C}} \end{bmatrix} \tag{2.6}$$

A theorem for global observability of this problem can be stated as follows

Theorem 2.1.1. *Assume that $p(\cdot)$ represents an orthographic projection. Given $\boldsymbol{s}_{\mathcal{I}_i}^{p_j}$ and $\boldsymbol{x}_{\mathcal{G}}^{r_i}$ for $M = 4$ non-coplanar point features and $N = 3$ distinct keyframes such that the $\boldsymbol{x}_{\mathcal{L}}^{\mathcal{C}_i}$ are not collinear and the $\boldsymbol{x}_{\mathcal{G}}^{r_i}$ are not collinear, the similarity transform between $\mathcal{G}$ and $\mathcal{L}$ and the quantities $\boldsymbol{x}_{\mathcal{L}}^{\mathcal{C}_i}$, $\boldsymbol{q}_{\mathcal{C}_i}^{\mathcal{L}}$, and $\boldsymbol{x}_{\mathcal{L}}^{p_j}$ for $i = 1, 2, 3$ and $j = 1, 2, 3, 4$ can be uniquely determined.*

To prove Theorem 2.1.1, consider the structure from motion (SFM) theorem given as

> Given three distinct orthographic projections of four non-coplanar points in a rigid configuration, the structure and motion compatible with the three views are uniquely determined up to a reflection about the image plane [35].

The reflection about the image plane can be discarded, as it exists behind the camera. Thus, the SFM theorem states that a unique solution for $\mathbf{x}_{\mathcal{L}}^{\mathcal{C}_i}$, $\mathbf{q}_{\mathcal{C}_i}^{\mathcal{L}}$, and $\mathbf{x}_{\mathcal{L}}^{p_j}$ can be found using only $\mathbf{s}_{\mathcal{I}_i}^{p_j}$ for $i = 1, 2, 3$ and $j = 1, 2, 3, 4$. The SFM theorem was proven by Ullman using a closed-form solution procedure [35].

The remainder of Theorem 2.1.1 is proven using the closed-form solution for finding a similarity transformation presented by Horn in [36]. Horn demonstrated that the similarity transform between two coordinate systems can be uniquely determined based on knowledge of the location of three non-collinear points in both coordinate systems. In the case of Theorem 2.1.1, this result allows the similarity transform between $\mathcal{G}$ and $\mathcal{L}$ to be recovered from the three locations of the reference point in the two frames, since the locations $\mathbf{x}_{\mathcal{G}}^{r_i}$ for $i = 1, 2, 3$ are given and the reference points $\mathbf{x}_{\mathcal{L}}^{r_i}$ can be computed from

$$\mathbf{x}_{\mathcal{L}}^{r_i} = \mathbf{x}_{\mathcal{L}}^{\mathcal{C}_i} + R(\mathbf{q}_{\mathcal{C}_i}^{\mathcal{L}})^T \mathbf{x}_{\mathcal{C}}^{r}, \text{ for } i = 1, 2, 3 \tag{2.7}$$

2.1.3 Interpretation of Results

Theorem 2.1.1 provides a sufficient condition for global observability of the locations of the point features and the pose of the camera in $\mathcal{G}$. This demonstrates that absolute pose can be recovered from the coupling of GPS, which provides the measurements of $\mathbf{x}_{\mathcal{G}}^{r_i}$, and visual SLAM in spite of neither being capable of determining absolute attitude alone. Interestingly, this means that an AR system that fully couples GPS and visual SLAM does not need to rely on an IMU for absolute attitude. This system would therefore not be susceptible to disturbances in the magnetic field, which can cause large pointing errors in the magnetometers in IMUs.

While the conditions specified in Theorem 2.1.1 are sufficient, they are certainly not necessary. Ullman mentions in his proof of the SFM theorem that under certain circumstances a unique solution still exists even if the four point features are coplanar [35]. The inclusion of GPS measurements may also have an effect on the required conditions for observability. While the weak local observability results from Sec. 2.1.1 do not prove the existence of a globally unambiguous solution, the results suggest that it may be possible to get by with just three point features. However, this thesis employs visual SLAM algorithms that track hundreds or even thousands of points, so specifying the absolute minimum conditions under which a solution exists is not of concern for this thesis.

2.2 Estimation Methods

The optimal approach to any causal estimation problem would be to gather all the measurements collected up to the current time and produce an estimate of the state from this entire batch by minimizing a cost function whenever a state estimate is desired [37]. The most commonly employed cost function is the weighted square of the measurement error in which case the estimation procedure is referred to as least-squares. In the case of linear systems, the batch least-squares estimation procedure simply involves gathering the measurements into a single matrix equation and performing a generalized matrix inversion [38]. In the case of nonlinear systems, the batch least-squares estimation procedure is somewhat more involved. Computation of the nonlinear least-squares solution typically involves linearization of the measurements about the current best estimate of the state, performing a generalized matrix inversion, and iteration of the procedure until the estimate settles on a minimum of the cost function [38]. While this approach is optimal, it often becomes enormously computationally intensive as more measurements are gathered and is thus often impractical for real-time applications.

This issue led to the development of the Kalman filter [39, 40], which is also optimal for linear systems where all noises are white and Gaussian distributed. The Kalman filter is a sequential estimation method that summarizes the information gained up to the current time as a multivariate Gaussian probability distribution. This development eliminated the need to process all the measurements at once, thus providing a more computationally-efficient process for real-time estimation.

The use of the Kalman filter was later extended to nonlinear systems by linearizing the system about the current best estimate of the state, as was done for the batch solution procedure. This method was coined the extended Kalman filter

(EKF). However, errors in the linearization applied by the EKF cause the filter to develop a bias and make the filter sub-optimal [41]. Iteration over the measurements within a certain time window can be performed to reduce the resulting bias without resorting to a batch process over all the measurements [41]. However, it is typically assumed that the linearization is close enough that these errors are small and this small bias is acceptable in order to enable real-time estimation. Non-Gaussianity can also be a problem with EKFs due to propagation of the distribution through non-linear functions. Other filtering methods have also been developed to better handle issues of non-Gaussianity caused by nonlinearities [42, 43].

As explained previously, batch estimation methods are typically dismissed in favor of sequential methods for real-time application because of the inherent computational expense of batch solutions. However, the unique nature of visual SLAM makes batch estimation appealing even for real-time application [44]. These unique aspects of the visual SLAM problem are

High Dimensionality: The images on which visual SLAM operates inherently have high dimensionality. Each image has hundreds or thousands of individual features that can be identified and tracked between images. These tracked features each introduce their own position as parameters that must be estimated in order for the features to be used for navigation. If all of the hundreds or thousands of image features from all the images in a video stream are to be used for navigating, then the problem quickly becomes infeasible for real-time applications based on computational requirements even for a sequential estimation method. Therefore, compromises must be made regarding either the number of features tracked, the frame rate, or both. This compromise is different for batch and sequential estimators; this point will be explained in detail in Sec. 2.2.1.

Inherent Sparsity: Linearized measurement equations in the visual SLAM problem have a banded structure in the columns corresponding to the feature locations when measurements taken from multiple frames are processed together. Sparse matrix structures such as this result in drastic computational savings when properly exploited. This inherent sparsity collapses if one tries to summarize data as in a recursive estimator.

Superfluity of Dynamic Constraints: While dynamic constraints on the camera poses from different frames do provide information to aid in estimation, this additional information is unnecessary for visual SLAM and may not be as valuable as preserving sparsity. Removing these dynamic constraints creates a block diagonal structure in the linearized measurement equations for a batch estimator in the columns corresponding to the camera poses. This sparse structure can be exploited by the batch estimator for additional computational savings. Thus, more features can be tracked by the batch estimator for the same computational expense by ignoring dynamic constraints.

Spatial Correlation: Since visual features must be in view of the camera to be useful for determining the current camera pose, past images that no longer contain visual features currently in view of the camera provide little or no information about the current camera pose. Thus, the images with corresponding camera poses and features that are not in the neighborhood of the current camera pose can be removed from the batch estimation procedure, reducing the size of both the state vector and the measurement vector in a batch solution procedure.

2.2.1 Methodologies for Visual SLAM

Two primary methodologies have been applied to the visual SLAM problem; each addresses the constraint of limited computational resources in fundamentally different ways. These methodologies are filter-based visual SLAM and bundle-adjustment-based visual SLAM. Each of these methods and the concessions made to reduce their computational expense are described below.

2.2.1.1 Filter-Based Visual SLAM

Filter-based visual SLAM employs a sequential-type estimator that marginalizes out past camera poses and the corresponding feature measurements by summarizing the information gained as a multi-variate probability distribution (typically Gaussian) of the current pose. For most problems, this marginalization of past poses maintains a small state vector and prevents the computational cost of the filter from growing. This is not the case for visual SLAM where each image could add many new features whose location must be estimated and maintained in the state vector.

Typical filter-based visual SLAM algorithms have computational complexity that is cubic with the number of features tracked due to the need for adding the feature locations to the state vector and propagating the state covariance through the filter [37]. To reduce computational expense, filter-based visual SLAM imposes limits on the number of features extracted from the images, thus preventing the state vector from becoming too large. Examples of implementations of filter-based visual SLAM can be found in [23–26].

Mourikis Method Of the filter-based visual SLAM methods reported in literature, the method designed by Mourikis et al. [23] is of particular interest. Mourikis created a measurement model for the feature measurements that expresses these measurements in terms of constraints on the camera poses for multiple images or frames. This linearized measurement model for a single feature over multiple frames is expressed as

$$\begin{aligned}\mathbf{z}_{p_j} = \mathbf{s}^{p_j} - \hat{\mathbf{s}}^{p_j} &= \left.\frac{\partial \mathbf{s}^{p_j}}{\partial \mathbf{X}}\right|_{\bar{\mathbf{X}},\bar{\mathbf{x}}^{p_j}} \delta\mathbf{X} + \left.\frac{\partial \mathbf{s}^{p_j}}{\partial \mathbf{x}^{p_j}}\right|_{\bar{\mathbf{X}},\bar{\mathbf{x}}^{p_j}} \delta\mathbf{x}^{p_j} + \mathbf{w}_{p_j} \\ &= H_{p_j,X}\delta\mathbf{X} + H_{p_j,x^{p_j}}\delta\mathbf{x}^{p_j} + \mathbf{w}_{p_j}\end{aligned} \tag{2.8}$$

where $\mathbf{s}^{p_j}$ is formed by stacking the feature measurements $\mathbf{s}_{\mathtt{J}_i}^{p_j}$ from Eq. 2.2 for each frame being processed, $\mathbf{X}$ is the state vector which includes the camera poses for the frames being processed, $\hat{\mathbf{s}}^{p_j}$ is the expected value of the feature measurements based on the a priori state $\bar{\mathbf{X}}$, $\delta\mathbf{X}$ and $\delta\mathbf{x}^{p_j}$ are the errors in the a priori state and feature location respectively, and $\mathbf{w}_{p_j}$ is white Gaussian measurement noise with a diagonal covariance matrix. The estimate of the feature location $\mathbf{x}^{p_j}$ is simply computed from the feature measurements and camera pose estimates from other frames that were not used in Eq. 2.8, but have already been collected and added to the state.

The measurement model in Eq. 2.8, however, still contains the error in the estimated feature locations. To obtain a measurement model that contains only the error in the state, Mourikis transformed Eq. 2.8 by left multiplying by a matrix, $A_{p_j}^T$, that spans the left null space of $H_{p_j,x^{p_j}}$ to obtain

$$A_{p_j}^T \mathbf{z}_{p_j} = \mathbf{z}'_{p_j} = H'_{p_j,X}\delta\mathbf{X} + \mathbf{w}'_{p_j} \tag{2.9}$$

This operation reduces the number of equations from $2N_f$, where N_f is the number of frames used in Eq. 2.8, to $2N_f - 3$, since the rank of $H_{p_j,x^{p_j}}$ is 3. This assumes that

$N_f > 1$, since the null space of $H_{p_j,x^{p_j}}$ would be empty otherwise. The remaining 3 equations, which are thrown out, are of the form

$$H^T_{p_j,x^{p_j}} \mathbf{z}_{p_j} = \mathbf{z}^r_{p_j} = H^r_{p_j,X} \delta \mathbf{X} + H^r_{p_j,x^{p_j}} \delta \mathbf{x}^{p_j} + \mathbf{w}^r_{p_j} \tag{2.10}$$

Since no guarantee can be made that $H^r_{p_j,X}$ in Eq. 2.10 will be zero, this procedure sacrifices information about the state by ignoring these 3 equations.

Therefore, the Mourikis implementation does not require the feature positions to be added to the state, but requires a limited number of camera poses to be added to the state instead. Once a threshold on the number of camera poses in the state is reached, a third of the camera poses are marginalized out of the state after processing the feature measurements associated with those frames using Eq. 2.9. This approach has computational complexity that is only linear with the number of features, but is cubic with the number of camera poses in the state. The number of camera poses maintained in the state can be made much smaller than the number of features, so this method is significantly more computationally efficient than traditional filter-based visual SLAM. Thus, this method allows more features to be tracked than with traditional filter-based visual SLAM for the same computational expense.

Modified Mourikis Method The Mourikis method has the undesirable qualities that (1) it throws away information that could be used to improve the state estimate and (2) the measurement update cannot be performed on a single frame. These drawbacks can be eliminated by recognizing that the feature locations are simply functions of the camera poses from the state in this method. This means that the error in the feature location can be expressed as

$$\delta \mathbf{x}^{p_j} = \left. \frac{\partial \mathbf{x}^{p_j}}{\partial \mathbf{X}} \right|_{\bar{\mathbf{X}}} \delta \mathbf{X} \tag{2.11}$$

These partial derivatives are quite complex and may need to be computed numerically. This allows the measurement equations to be expressed entirely in terms of the state vector by substituting Eq. 2.11 into Eq. 2.8, so no information needs to be discarded and the measurement update can be performed using a single frame.

This modified version of the Mourikis method has a state vector that can be partitioned into two sections. The first portion of the state contains the current camera pose. The second portion of the state contains the camera poses for frames that are specially selected to be spatially diverse. These specially selected frames are referred to as keyframes.

Measurements from the keyframes are used to compute the estimates of the feature locations and are not processed by the filter. The estimates of the feature locations can be updated in a thread separate from the filter whenever processing power is available using the current best estimate of the keyframe poses from the state vector. New features are also identified in the keyframes as allowed by available processing power. This usage of keyframes is inspired by the bundle-adjustment-based visual SLAM algorithm developed by Klein and Murray [45], which will be detailed in Sec. 2.2.1.2.

When a new frame is captured, this method first checks if this frame should be added to the list of keyframes. If so, then the current pose is appended to the end of the state vector and the measurements from the frame are not processed by the filter. Otherwise, the linearized measurement equations are formed from Eqs. 2.8 and 2.11 and used to update the state.

To prevent the number of keyframes from growing without bound, the keyframes are removed from the state whenever the system is no longer in the neighborhood where the keyframe was taken. This condition can be detected by a set of heuristics that compare the keyframe pose and the current pose of the system to see if the two are still close enough to keep the keyframe in the state. When a keyframe is removed, the current best estimate and covariance of the associated pose and the associated measurements can be saved for later use. If the system returns to the neighborhood again, then the keyframes from that neighborhood can be reloaded into the state. This should enable loop closure, which most visual SLAM implementations have difficulty accomplishing.

2.2.1.2 Bundle-Adjustment-Based Visual SLAM

Bundle-adjustment-based visual SLAM, in contrast to filter-based visual SLAM, does not marginalize out the past poses. Bundle Adjustment (BA) is a batch nonlinear least-squares algorithm that collects measurements of features from all of the frames collected and processes them together. Implementing this process as a batch solution allows the naturally sparse structure of the visual SLAM problem to be exploited and eliminates the need to compute state covariances. This allows BA to obtain computational complexity that is linear in the number of features tracked [44, 46].

This approach is optimal, but computing global BA solutions for visual SLAM is a computationally intensive process that cannot be performed at the frame-rate of the camera. As such, BA-based visual SLAM only selects certain “keyframes” to incorporate into the global BA solution, which is computed only occasionally or as processing power is available [37]. Pose estimates for each frame can then be computed directly using the feature positions obtained from the global BA solution and the measured feature coordinates in the image. BA-based visual SLAM typically does not compute covariances, which are not required for BA and would increase the computational cost significantly.

Parallel Tracking and Mapping The predominant BA-based visual SLAM algorithm was developed by Klein and Murray [45] and is called parallel tracking and mapping (PTAM). PTAM is capable of tracking thousands of features and estimating relative pose up to an arbitrary scale-factor at 30 Hz frame-rates on a dual-core computer. PTAM is divided into two threads designed to operate in parallel. The first thread is the mapping thread, which performs BA to compute a map of the environment and identifies new point features in the images. The second thread is the tracking thread, which identifies point features from the map in new frames, computes the camera pose for the new frames, and determines if new frames should be added to the list of keyframes or discarded. PTAM is only designed to operate in small workspaces, but can be adapted to larger workspaces by trimming the map in the same way described for the modified Mourikis method in Sec. 2.2.1.1.

2.2.2 Optimal Approach for Visual SLAM

The two methodologies for visual SLAM, filter-based and BA-based, have been discussed, but the question remains as to which approach gives the best performance for visual SLAM. Filter-based visual SLAM has the advantage of processing every camera frame, but imposes severe limits on the number of point features tracked due to cubic computational complexity. The modified Mourikis method attains linear computational complexity with the number of tracked features, but has cubic computational complexity with the number of poses in the state. Filter-based methods also suffer from linearization errors during the marginalization of frames. BA-based

Table 2.1: Ranking of Visual SLAM Methodologies

Estimator Type	Methodology	Accuracy	Robustness	Computational Efficiency
Batch	Bundle Adjustment	1	1	1
Sequential	Traditional SLAM	3	3	3
	Modified Mourikis	2	2	2

visual SLAM has several advantages over filter-based visual SLAM including linear computational complexity in the number of tracked features and the elimination of linearization errors through iteration over the entire set of data, but must reduce the number of frames incorporated into the batch processing to achieve real-time operation.

Strasdat et al. performed a comparative analysis of the performance of both visual SLAM methodologies which revealed that BA-based visual SLAM is the optimal choice based on the metric of accuracy per computational cost [37]. The primary argument that Strasdat et al. present was that accuracy is best increased by tracking more features. Their results demonstrated that after adding a few keyframes from a small region of operation only extremely marginal benefit was obtained by adding more frames. Based on this fact, BA was able to obtain better accuracy per computational cycle than the filter due to the difference in computational complexity with the number of features tracked. Strasdat et al. did not consider any method like the modified Mourikis method in their analysis, which would have significant improvements in accuracy per computational cost over traditional filter-based methods. However, there is no reason to expect the modified Mourikis method would outperform BA. To summarize this analysis, Table 2.1 shows a ranking of these methods for the metrics of accuracy, robustness, and computational efficiency.

2.2.3 Methodologies for Coupled Visual SLAM, GPS, and Inertial Sensors

Now consider adding GPS and inertial measurements to the visual SLAM problem. The addition of GPS measurements links the pose estimate to a global coordinate system, as proven in Sec. 2.1. Inertial measurements from a three-axis accelerometer and a three-axis gyro help to smooth out the solution between measurement updates and limit the drift of this global reference during periods when GPS is unavailable.

Although BA proved to be the optimal method for visual SLAM alone, this may not be the case for combined visual SLAM, GPS, and inertial sensors. Filtering is generally the preferred technique for navigating with GPS and inertial sensors for good reason. Inertial measurements are typically collected at a rate of 100 Hz or greater to accurately reconstruct the dynamics of the system between measurements. Taking inertial measurements much less frequently would defeat the purpose of having the measurements, so they should not be ignored to reduce the number of measurements. The matrices resulting from a combined GPS and inertial sensors navigation system are also not sparse like in visual SLAM, so the computational efficiency associated with sparseness cannot be exploited. This means that a solely batch estimation algorithm is computationally infeasible for this problem. Therefore, a hybrid batch-sequential or entirely sequential method that obtains high accuracy and robustness with low computational cost is desired.

2.2.3.1 Coupled Bundle-Adjustment-Based Visual SLAM, GPS, and Inertial Sensors

One potential method for coupling these navigation techniques is to process the keyframes using BA and process the measurements from the other frames, GPS, and inertial sensors through a filter without adding the feature locations to the filter state. Specifically, BA would estimate the feature locations and keyframe poses based on the visual feature measurements from the keyframes and a priori keyframe pose estimates provided by the filter. Adding these a priori keyframe pose estimates to the BA cost function does not destroy sparseness because the a priori keyframe poses are represented as independent from one another. The BA solution for the feature locations will also be expressed in the same global reference frame as the a priori keyframe pose estimates. The filter would process all GPS measurements in a standard fashion and use the inertial measurements to propagate the state forward in time between measurements. Frames not identified as keyframes would also be processed by the filter using the estimated feature locations from BA.

An important detail in this approach is precisely how the feature locations from BA are used to process the non-keyframes in the filter. Using the BA estimated feature locations in the filter measurement equations without representing their covariance will cause issues with the filter covariance estimate being overly optimistic. This overly optimistic covariance will then feed back into BA whenever a new keyframe is added and could cause divergence of the estimated pose. This is clearly unacceptable, so the covariance of the estimated feature locations should be computed for use in the filter. However, computing this covariance matrix can only be done at considerable computational expense, which cuts against the main benefit of using BA. To reduce the computational load of computing these covariance matrices in BA, the covariance matrix of each individual feature may be computed efficiently by ignoring cross-covariances between camera poses and other features. This approximation will be somewhat optimistic, but this could be accounted for by slightly inflating the measurement noise.

By separating the estimation of the feature locations and keyframe poses from the filter, the coupling between the current state, keyframe poses, and feature measurements is not fully represented. The estimator essentially ignores the cross-covariances between these quantities. This prevents GPS and IMU measurements from aiding BA, except by providing a better a priori estimate of the keyframe poses. While this feature of the estimator is undesirable, it may not significantly degrade performance.

2.2.3.2 Coupled Filter-Based Visual SLAM, GPS, and Inertial Sensors

Another approach to this problem would be to transition entirely to a filter implementation, which allows full exploitation of the coupling between the states. One could implement this approach using either the traditional visual SLAM approach or the modified Mourikis method for visual SLAM presented in Sec. 2.2.1.1. The filter would process all GPS measurements in a standard fashion and use the inertial measurements to propagate the state forward in time between measurements. However, the traditional visual SLAM approach has no benefits over the modified Mourikis method and has much greater computational cost, so there is no advantage to considering it here.

2.2.4 Optimal Approach for Coupled Visual SLAM, GPS, and Inertial Sensors

Table 2.2 shows an incomplete ranking of a full batch solution, the hybrid batch-sequential method employing BA for visual SLAM, and the entirely sequential approach employing the modified Mourikis method for visual SLAM. While the computational complexity for all the methods is known, the accuracy and robustness of the two proposed methods are unknown at this time. The hybrid method using BA

Table 2.2: Ranking of Combined Visual SLAM, GPS, and Inertial Sensors Methodologies

Estimator Type	Methodology	Accuracy	Robustness	Computational Efficiency
Batch	Full Batch	1	1	3
Hybrid	BA SLAM + Filter	?	?	1
Sequential	Modified Mourikis	?	?	2

has the advantage of being able to track more features and maintain more keyframes for the same computational cost compared to the sequential method, though this advantage is somewhat diminished by the need to compute a covariance matrix. On the other hand, the hybrid method does not represent the coupling between the current state, the keyframe poses, and the feature locations and thus sacrifices this information for computational efficiency. The sequential method properly accounts for this coupling.

It is difficult to tell which method will perform better for the same computational cost without implementing and testing these methods. The remainder of this thesis presents a navigation filter and prototype AR system that implements a looser coupling of these navigation techniques as a first step towards the goal of implementing the methodologies discussed in this chapter.

Chapter 3

Measurement and Dynamics Modeling

Assuming a mobile AR system with internet access is given that rigidly connects a GPS receiver, a camera, and an IMU, a navigation system estimating absolute pose of the AR system can be designed that couples CDGPS, visual SLAM, and an INS. Potential optimal strategies for fusing measurements from these navigation techniques were discussed previously in Sec. 2.2.3. These strategies, however, all require a tighter coupling of the visual SLAM algorithm with the GPS observables and inertial measurements than can be obtained using stand-alone visual SLAM software. Thus, these methods necessitate creation of a new visual SLAM algorithm or significant modification to an existing stand-alone visual SLAM algorithm. In keeping with a staged developmental approach, the prototype system whose results are reported in this thesis implements a looser coupling of the visual SLAM algorithm with the GPS observables and inertial measurements. In particular, this thesis instead considers a navigation filter that employs GPS observables measurements, IMU accelerometer measurements and attitude estimates, and relative pose estimates from a stand-alone visual SLAM algorithm. While this implementation does not allow the navigation system to aid visual SLAM, it still demonstrates the potential of such a system for highly-accurate pose estimation. Additionally, the accuracy of both globally-referenced position and attitude are improved over a coupled CDGPS and INS navigation system through the incorporation of visual SLAM in this framework.

This chapter introduces measurement and dynamics models that are used in creating a navigation filter in Chapter 4. The chapter begins by providing an overview of the navigation system developed in this thesis that includes a block diagram of the overall system and the definition of the state vector of the filter. Next, the measurement models for the GPS observables, IMU accelerometer measurements and attitude estimates, and visual SLAM relative pose estimates are derived and linearized about the filter state. Finally, the dynamics models of the system both with and without accelerometer measurements from the IMU are presented.

3.1 Navigation System Overview

The navigation system presented in this thesis is an improved version of that presented in [47]. This prior version of the system did not incorporate visual SLAM measurements nor did it represent attitude estimates properly in the filter.

3.1.1 Reference Frames

The navigation system developed in this thesis utilizes five different reference frames. These reference frames are

Earth-Centered, Earth-Fixed (ECEF) Frame: This is one of the standard global reference frames whose origin is at the center of the Earth and rotates with the Earth.

East, North, Up (ENU) Frame: This reference frame is defined by the local east, north, and up directions which can be determined by simply specifying a location in ECEF as the origin of the frame.

Camera (C) Frame: A reference frame that is centered on the focal point of the camera with the z-axis pointing down the bore-sight of the camera, the x-axis pointing toward the right in the image frame, and the y-axis completing the right-handed triad.

Body (B) Frame: A reference frame that is centered at a point on the AR system and rotates with the AR system. This reference frame is assigned differently based on the types measurements employed by the filter. When INS measurements are present, this frame is centered on the IMU origin and aligned with the axes of the IMU to simplify the dynamics model given in Sec. 3.3.1. If there are visual SLAM measurements and no INS measurements, then this frame is the same as the camera frame. This is the most sensible definition of the body frame, since estimating the camera pose is the goal of this navigation filter. If only GPS measurements are present, then this frame is centered on the phase center of the mobile GPS antenna because attitude cannot be determined by the system.

Vision (V) Frame: A reference frame arbitrarily assigned by the visual SLAM algorithm during initialization. The vision frame is related to ECEF by a constant, but unknown, similarity transform — a combination of translation, rotation, and scaling.

3.1.2 Block Diagram of the Navigation System

A block diagram of the navigation system is shown in Fig. 3.1. This block diagram identifies the subsystems within the navigation system as a whole by encircling the corresponding blocks with a colored dashed line. These colors are red for the INS, blue for CDGPS, and green for the visual navigation system (VNS). The navigation filter is responsible for combining the measurements from these independent subsystems to estimate the state of the AR system. Blocks within the navigation filter are

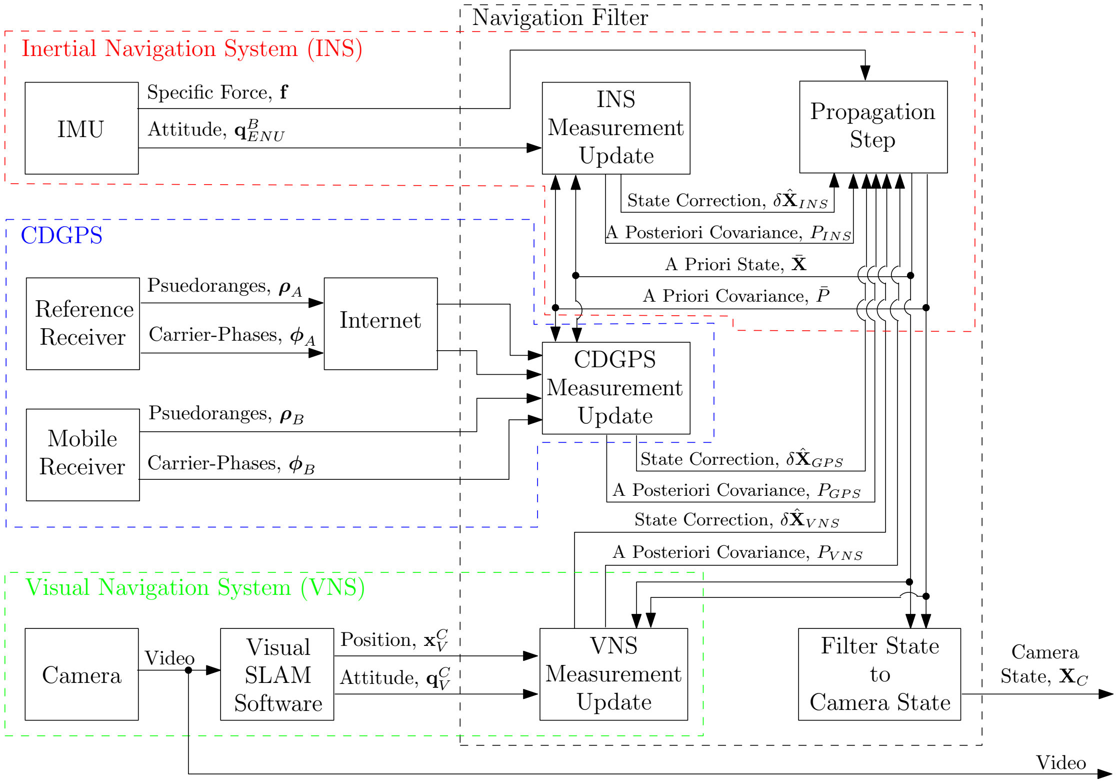
Navigation Filter
Inertial Navigation System (INS)
IMU
Specific Force, $\mathbf{f}$
Attitude, $\mathbf{q}^B_{ENU}$
INS Measurement Update
Propagation Step
State Correction, $\delta\hat{\mathbf{X}}_{INS}$
A Posteriori Covariance, P_{INS}
A Priori State, $\bar{\mathbf{X}}$
A Priori Covariance, $\bar{P}$
CDGPS
Reference Receiver
Psuedoranges, $\boldsymbol{\rho}_A$
Carrier-Phases, $\boldsymbol{\phi}_A$
Internet
CDGPS Measurement Update
Mobile Receiver
Psuedoranges, $\boldsymbol{\rho}_B$
Carrier-Phases, $\boldsymbol{\phi}_B$
State Correction, $\delta\hat{\mathbf{X}}_{GPS}$
A Posteriori Covariance, P_{GPS}
State Correction, $\delta\hat{\mathbf{X}}_{VNS}$
A Posteriori Covariance, P_{VNS}
Visual Navigation System (VNS)
Camera
Video
Visual SLAM Software
Position, $\mathbf{x}^C_V$
Attitude, $\mathbf{q}^C_V$
VNS Measurement Update
Filter State to Camera State
Camera State, $\mathbf{X}_C$
Video

Figure 3.1: A block diagram of the navigation system.

encircled by a black dashed line. The sensors for the system are all aligned in a single column on the far left side of Fig. 3.1. The outputs from the navigation system are the state of the camera, which includes the absolute pose, and the video from the camera.

This type of navigation system can be implemented on a large scale with minimal infrastructure. The required sensors for this navigation system are all located on the AR system, except for the reference receiver, and none of the sensors require the area of operation to be prepared in any way. The reference receiver is a stationary GPS receiver at a known location that provides GPS observables measurements to the system via the internet. A single reference receiver can provide measurements to an unlimited number of systems at distances as large as 10 km away from the reference receiver for single-frequency CDGPS and even further for dual-frequency CDGPS. This means that only a sparsely populated network of reference receivers is required to service an unlimited number of navigation systems similar to this one over a large area.

The navigation system developed in this thesis has several modes of operation depending on what measurements are provided to it. These modes are CDGPS-only, CDGPS and INS, CDGPS and VNS, and CDGPS, VNS, and INS. This allows testing and comparison of the performance of the different subsystems. Whenever measurements from a subsystem are not present, the portion of the block diagram corresponding to that subsystem shown in Fig. 3.1 is removed and the state vector is modified to remove any states specific to that subsystem. In the case that INS measurements are not present, the propagation step block is modified to use an INS-free dynamics model instead of being entirely removed.

3.1.3 State Vector of the Navigation System

A typical CDGPS navigation filter has a state of the form

$$\mathbf{X}_{CDGPS} = \begin{bmatrix} (\mathbf{x}^{B}_{ECEF})^T & (\mathbf{v}^{B}_{ECEF})^T & (\mathbf{N})^T \end{bmatrix}^T \tag{3.1}$$

where $\mathbf{x}^{B}_{ECEF}$ and $\mathbf{v}^{B}_{ECEF}$ are the position and velocity of the origin of the B-frame in ECEF and $\mathbf{N}$ is the vector of CDGPS carrier-phase integer ambiguities. The carrier-phase integer ambiguities are constant and arise as part of the CDGPS solution, which is described in detail in Sec. 3.2.1.

Adding an INS that provides accelerometer measurements and attitude estimates to the CDGPS navigation filter necessitates the addition of the accelerometer bias, $\mathbf{b}_a$, and the attitude of the B-frame relative to ECEF, $\mathbf{q}^B_{ECEF}$, to the state. The resulting state for coupled CDGPS and INS is

$$\mathbf{X}_{CDGPS/INS} = \left[\begin{array}{ccccc} (\mathbf{x}^B_{ECEF})^T & (\mathbf{v}^B_{ECEF})^T & (\mathbf{b}_a)^T & (\mathbf{q}^B_{ECEF})^T & (\mathbf{N})^T \end{array} \right]^T \tag{3.2}$$

If, instead of an INS, a VNS that provides relative pose estimates in some arbitrary V-frame is coupled to the CDGPS filter, then the constant similarity transform between the V-frame and ECEF must be added to the state in addition to the attitude of the B-frame relative to ECEF. The need for the arbitrarily assigned V-frame could be eliminated if the navigation filter provided the VNS with estimates of the absolute pose at each camera frame, as shown in Sec. 2.1, but this is not the case for the navigation system presented in this thesis. The resulting state for coupled CDGPS and VNS is

$$\begin{aligned} &\mathbf{X}_{CDGPS/VNS} \\ &= \left[\begin{array}{ccccccc} (\mathbf{x}^B_{ECEF})^T & (\mathbf{v}^B_{ECEF})^T & (\mathbf{q}^B_{ECEF})^T & (\mathbf{x}^V_{ECEF})^T & (\mathbf{q}^{ECEF}_V)^T & \lambda & (\mathbf{N})^T \end{array} \right]^T \end{aligned} \tag{3.3}$$

where $\mathbf{x}^V_{ECEF}$, $\mathbf{q}^{ECEF}_V$, and λ are the translation, rotation, and scale-factor respectively which parameterize the similarity transform relating the V-frame and ECEF.

The state vector for the full navigation filter that couples CDGPS, VNS, and INS is obtained by adding the accelerometer bias to the state for coupled CDGPS and VNS from Eq. 3.3. This results in

$$\begin{aligned} \mathbf{X} &= \mathbf{X}_{CDGPS/VNS/INS} \\ &= \left[\begin{array}{cccccccc} (\mathbf{x}^B_{ECEF})^T & (\mathbf{v}^B_{ECEF})^T & (\mathbf{b}_a)^T & (\mathbf{q}^B_{ECEF})^T & (\mathbf{x}^V_{ECEF})^T & (\mathbf{q}^{ECEF}_V)^T & \lambda & (\mathbf{N})^T \end{array} \right]^T \end{aligned} \tag{3.4}$$

This state vector will be used throughout the remainder of this thesis. It should be noted that the models for the other modes of the navigation filter, CDGPS-only, CDGPS and INS, and CDGPS and VNS, can be obtained from the models for the full navigation filter by simply ignoring the terms in the linearized models corresponding to states not present in that mode's state vector.

Each of the state vectors can be conveniently partitioned to obtain

$$\mathbf{X} = \begin{bmatrix} \mathbf{x} \\ \mathbf{N} \end{bmatrix} \tag{3.5}$$

where $\mathbf{x}$ contains the real-valued part of the state and $\mathbf{N}$ contains the integer-valued portion of the state, which is simply the vector of CDGPS carrier-phase integer ambiguities. This partitioning of the state will be used throughout the development of the filter, since it is convenient for solving for the state after measurement updates.

Attitude of both the AR system and the V-frame is represented using quaternions in the state vector. Quaternions are a non-minimal attitude representation that is constrained to have unit norm. To enforce this constraint in the filter, the quaternions $\mathbf{q}^{B}_{ECEF}$ and $\mathbf{q}^{ECEF}_{V}$ are replaced in the state with a minimal attitude representation, denoted as $\delta\mathbf{e}^{B}_{ECEF}$ and $\delta\mathbf{e}^{ECEF}_{V}$ respectively, during measurement updates and state propagation [48]. This is accomplished through the use of differential quaternions, which are described in detail in Appendix A. These differential quaternions represent a small rotation from the current attitude to give an updated estimate of the attitude through the equation

$$\mathbf{q}' = \delta\mathbf{q}(\delta\mathbf{e}) \otimes \mathbf{q} \tag{3.6}$$

where $\mathbf{q}'$ is the updated attitude estimate and $\delta\mathbf{q}(\delta\mathbf{e})$ is the differential quaternion.

As a matter of notation, the state itself or elements of the state vector when substituted into models will be denoted with either a bar, $\bar{(\cdot)}$, for a priori estimates or a hat, $\hat{(\cdot)}$, for a posteriori estimates. Any term representing the state or an element of the state without these accents is the true value of that parameter. When the state or an element of the state has a delta in front of it, $\delta(\cdot)$, this represents a linearized correction term to the current value of the state. The same accent rules that apply to the state also apply to delta states.

3.2 Measurement Models

3.2.1 CDGPS Measurement Models

The signal tracking loops of a GPS receiver produce a set of three measurements, typically referred to as observables, which are used in computing the receivers position-velocity-time (PVT) solution. These observables are pseudorange, beat carrier-phase, and Doppler frequency. In SPS GPS, the pseudorange and Doppler frequency measurements are used to compute the position and velocity of the receiver respectively. The carrier-phase measurement, which is the integral of the Doppler frequency, is typically ignored or not even produced.

Carrier-phase can be measured to millimeter-level accuracy, but there exists an inherent range ambiguity that is difficult to resolve in general. CDGPS is a technique that arose to reduce the difficulty in resolving this ambiguity. This is accomplished by differencing the measurements between two receivers, a stationary reference receiver (RX A) and a mobile receiver (RX B), and between two satellites. The resulting measurements are referred to as double-differenced measurements. Differencing the measurements eliminates many of the errors in the measurements and results in integer ambiguities that can be determined much quicker than their real-valued counterparts by enforcing the integer constraint. The downside to this process is that only relative position between the antennas of the two receivers can be determined to centimeter-level or better accuracy. However, the reference receiver can be placed at a surveyed location so that its absolute position can be nearly perfectly known ahead of time. As such, the analysis presented in this thesis will assume that the coordinates of the reference receiver are known. Further information on the GPS measurement models and CDGPS in general can be found in [49–52].

The navigation filter forms double-differenced measurements for both pseudorange and carrier-phase measurements from the civil GPS signal at the L1 frequency. Differencing the pseudorange measurements is not strictly necessary, but simplifies the filter development and reduces the required state vector. Time alignment of the pseudorange and carrier-phase measurements from both receivers must be obtained to form the double-differenced measurements. It is highly unlikely that the receiver time epochs when the pseudorange and carrier-phase measurements are taken for both receivers would correspond to the same true time. Therefore, these measurements must be interpolated to the same time instant before the double-differenced measurements are formed. This is typically performed using the Doppler frequency and the SPS GPS time solution, which are already reported by the receivers.

3.2.1.1 Undifferenced pseudorange and Carrier-Phase Measurement Models

The undifferenced pseudorange and carrier-phase models for RX B are

$$\begin{aligned}\rho_B^i(k) = r_B^i(k) + c\left(\delta t_{RX_B}(k) - \delta t_{SV_i}(k)\right) + I_B^i(k) + T_B^i(k) \\ + M_B^i(k) + w_{\rho,B}^i(k)\end{aligned} \tag{3.7}$$

$$\begin{aligned}\lambda_{L1}\phi^i_B(k) = r^i_B(k) &+ c\left(\delta t_{RX_B}(k) - \delta t_{SV_i}(k)\right) + \lambda_{L1}(\gamma^i_B - \psi^i) \\ &- I^i_B(k) + T^i_B(k) + m^i_B(k) + w^i_{\phi,B}(k)\end{aligned} \tag{3.8}$$

where $\rho^i_B(k)$ and $\phi^i_B(k)$ are the pseudorange and carrier-phase measurements in meters and cycles respectively from RX B for the ith satellite vehicle (SV), $r^i_B(k)$ is the true range from RX B to the ith SV, c is the speed of light, $\delta t_{RX_B}(k)$ is the receiver clock offset for RX B, $\delta t_{SV_i}(k)$ is the satellite clock offset for the ith SV, $I^i_B(k)$ and $T^i_B(k)$ are the Ionosphere and Troposphere delays respectively, $M^i_B(k)$ and $m^i_B(k)$ are the multipath errors on the pseudorange and carrier-phase measurements respectively, λ_{L1} is the wavelength of the GPS L1 frequency, γ^i_B is the initial carrier-phase of the signal when the ith SV was acquired by RX B, ψ^i is the initial broadcast carrier-phase from the ith SV, and $w^i_{\rho,B}(k)$ and $w^i_{\phi,B}(k)$ are zero-mean Gaussian white noise on the pseudorange and carrier-phase measurements respectively. The model for RX A is identical to this one with the appropriate values referenced to RX A instead.

The true range to the ith SV from RX B can be written as

$$r^i_B(k) = \left|\left|\mathbf{x}^{SV_i}_{ECEF}(k) - \mathbf{x}^{RX_B}_{ECEF}(k)\right|\right| \tag{3.9}$$

where $\mathbf{x}^{SV_i}_{ECEF}(k)$ is the position of the ith SV at the time the signal was transmitted and $\mathbf{x}^{RX_B}_{ECEF}(k)$ is the position of the phase center of the GPS antenna at the time the signal was received. The position of the satellites can be computed from the broadcast ephemeris data on the GPS signal. The position of the phase center of the GPS antenna is related to the pose of the system through the equation

$$\mathbf{x}^{RX_B}_{ECEF}(k) = \mathbf{x}^{B}_{ECEF}(k) + R(\mathbf{q}^B_{ECEF}(k))\mathbf{x}^{GPS}_B \tag{3.10}$$

where $\mathbf{x}_B^{GPS}$ is the position of the phase center of the GPS antenna in the B-frame.

The standard deviation of the pseudorange and carrier-phase measurement noises depend on the configuration of the tracking loops of the GPS receiver and the received carrier-to-noise ratio of the signal. Based on a particular tracking loop configuration, these standard deviations can be expressed in terms of the standard deviation of the pseudorange and carrier-phase measurements for a signal at some reference carrier-to-noise ratio through the relations

$$E\left[\left(w_{\rho,B}^i(k)\right)^2\right] = \left(\sigma_{\rho,B}^i(k)\right)^2 = \sigma_\rho^2\left(\left(\frac{C}{N_0}\right)_{ref}\right)\left(\frac{(C/N_0)_{ref}}{(C/N_0)_B^i(k)}\right) \tag{3.11}$$

$$E\left[\left(w_{\phi,B}^i(k)\right)^2\right] = \left(\sigma_{\phi,B}^i(k)\right)^2 = \sigma_\phi^2\left(\left(\frac{C}{N_0}\right)_{ref}\right)\left(\frac{(C/N_0)_{ref}}{(C/N_0)_B^i(k)}\right) \tag{3.12}$$

where $(C/N_0)_{ref}$ is the reference carrier-to-noise ratio in linear units, $(C/N_0)_B^i(k)$ is the received carrier-to-noise ratio of the signal from the ith SV by RX B in linear units, and $\sigma_\rho\left((C/N_0)_{ref}\right)$ and $\sigma_\phi\left((C/N_0)_{ref}\right)$ are the standard deviations of the pseudorange and carrier-phase measurements respectively for the particular tracking loop configuration at the reference carrier-to-noise ratio. Reasonable values for $\sigma_\rho\left((C/N_0)_{ref}\right)$ and $\sigma_\phi\left((C/N_0)_{ref}\right)$ at a reference carrier-to-noise ratio of 50 dB-Hz are 1 m and 2.5 mm respectively. The standard deviation of the pseudorange and carrier-phase measurement noise for RX A follows this same relation assuming that the tracking loop configurations are the same. It should be noted that the pseudorange and carrier-phase measurements are only negligibly correlated with one another and they are not correlated between receivers or SVs.

3.2.1.2 Single-Differenced pseudorange and Carrier-Phase Measurement Models

The pseudorange and carrier-phase measurements from Eqs. 3.7 and 3.8 are first differenced between the two receivers. This requires that both receivers be tracking the same set of satellites, which may be a subset of the satellites tracked by each receiver alone. The resulting single-differenced measurements are modeled as

$$\begin{aligned}\Delta\rho^i_{AB}(k) &= \Delta r^i_{AB}(k) + c\left(\delta t_{RX_A}(k) - \delta t_{RX_B}(k)\right) \\ &\quad + \Delta M^i_{AB}(k) + \Delta w^i_{\rho,AB}(k)\end{aligned} \tag{3.13}$$

$$\begin{aligned}\lambda_{L1}\Delta\phi^i_{AB}(k) &= \Delta r^i_{AB}(k) + c\left(\delta t_{RX_A}(k) - \delta t_{RX_B}(k)\right) \\ &\quad + \lambda_{L1}(\gamma^i_A - \gamma^i_B) + \Delta m^i_{AB}(k) + \Delta w^i_{\phi,AB}(k)\end{aligned} \tag{3.14}$$

where the single-difference operator Δ is defined as

$$\Delta(\cdot)_{AB} = (\cdot)_A - (\cdot)_B \tag{3.15}$$

The single-differenced pseudorange and carrier-phase measurement noises are still independent zero-mean Gaussian white noises, but the standard deviation is now

$$E\left[\left(\Delta w^i_{\rho,AB}(k)\right)^2\right] = \left(\sigma^i_{\rho,AB}(k)\right)^2 = \left(\sigma^i_{\rho,A}(k)\right)^2 + \left(\sigma^i_{\rho,B}(k)\right)^2 \tag{3.16}$$

$$E\left[\left(\Delta w^i_{\phi,AB}(k)\right)^2\right] = \left(\sigma^i_{\phi,AB}(k)\right)^2 = \left(\sigma^i_{\phi,A}(k)\right)^2 + \left(\sigma^i_{\phi,B}(k)\right)^2 \tag{3.17}$$

Differencing these measurements between the two receivers eliminated several error sources in the measurements. First, the satellite clock offset was eliminated, since this is common to both measurements. This error can also be removed by computing the satellite clock offset from the broadcast ephemeris data on the GPS signal, although these estimates are not perfect. Second, Ionosphere and Troposphere delays were eliminated under the assumption that the two receivers are close enough to one another that the signal traveled through approximately the same portion of the atmosphere. This assumption is the primary limitation on the maximum distance between the two receivers. As this baseline distance increases and this assumption is violated, the performance of CDGPS degrades. For a single-frequency CDGPS algorithm, the maximum baseline for centimeter-level positioning accuracy is about 10 km. Dual-frequency CDGPS algorithms can estimate the ionospheric delay at each receiver and remove it independent of the baseline distance, which can increase this baseline distance limit significantly.

Another effect of performing this first difference is the elimination of the initial broadcast carrier-phase of the satellite. This was one of the contributing factors to the carrier-phase ambiguity. However, the ambiguity on the single-differenced measurements is still real-valued.

3.2.1.3 Double-Differenced pseudorange and Carrier-Phase Measurement Models

Of the satellites tracked by both receivers, one satellite is chosen as the “reference” satellite which is denoted with the index 0. The single differenced measurements from this reference satellite are subtracted from those from all other satellites tracked by both receivers to form the double-differenced measurements. These double-differenced measurements are modeled as

$$\begin{aligned}\nabla\Delta\rho^{i0}_{AB}(k) &= \Delta\rho^{i}_{AB}(k) - \Delta\rho^{0}_{AB}(k) \\ &= \nabla\Delta r^{i0}_{AB}(k) + \nabla\Delta M^{i0}_{AB}(k) + \nabla\Delta w^{i0}_{\rho,AB}(k)\end{aligned} \tag{3.18}$$

$$\begin{aligned}\lambda_{L1}\nabla\Delta\phi^{i0}_{AB}(k) &= \lambda_{L1}\left(\Delta\phi^{i}_{AB}(k) - \Delta\phi^{0}_{AB}(k)\right) \\ &= \nabla\Delta r^{i0}_{AB}(k) + \lambda_{L1}N^{i0}_{AB} + \nabla\Delta m^{i0}_{AB}(k) + \nabla\Delta w^{i0}_{\phi,AB}(k)\end{aligned} \tag{3.19}$$

where N^{i0}_{AB} are the carrier-phase integer ambiguities and the double-difference operator is defined as

$$\nabla\Delta(\cdot)^{ij}_{AB} = \Delta(\cdot)^{i}_{AB} - \Delta(\cdot)^{j}_{AB} \tag{3.20}$$

The double-differenced pseudorange and carrier-phase measurement noises are still zero-mean Gaussian white noises, but the standard deviation is now

$$E\left[\left(\nabla\Delta w^{i0}_{\rho,AB}(k)\right)^2\right] = \left(\sigma^{i0}_{\rho,AB}(k)\right)^2 = \left(\sigma^{i}_{\rho,AB}(k)\right)^2 + \left(\sigma^{0}_{\rho,AB}(k)\right)^2 \tag{3.21}$$

$$E\left[\left(\nabla\Delta w^{i0}_{\phi,AB}(k)\right)^2\right] = \left(\sigma^{i0}_{\phi,AB}(k)\right)^2 = \left(\sigma^{i}_{\phi,AB}(k)\right)^2 + \left(\sigma^{0}_{\phi,AB}(k)\right)^2 \tag{3.22}$$

This second difference also created cross-covariance terms given by

$$E\left[\nabla\Delta w^{i0}_{\rho,AB}(k)\nabla\Delta w^{j0}_{\rho,AB}(k)\right] = \left(\sigma^{i0,j0}_{\rho,AB}(k)\right)^2 = \left(\sigma^{0}_{\rho,AB}(k)\right)^2, \text{ for } i \neq j \tag{3.23}$$

$$E\left[\nabla\Delta w^{i0}_{\phi,AB}(k)\nabla\Delta w^{j0}_{\phi,AB}(k)\right] = \left(\sigma^{i0,j0}_{\phi,AB}(k)\right)^2 = \left(\sigma^{0}_{\phi,AB}(k)\right)^2, \text{ for } i \neq j \tag{3.24}$$

This suggests that the satellite with the lowest single-differenced measurement noise should be chosen as the reference satellite to minimize the double-differenced measurement covariance.

Taking this second difference had two primary effects on the measurements. First, the receiver clock bias for both receivers was eliminated, since the biases are common to all single-differenced measurements. This means that the receiver clock biases no longer need to be estimated by the filter. Second, the ambiguities on the carrier-phase measurements are now integer-valued. This simplification only occurs if the receivers are designed such that the beat carrier-phase measurement is referenced to the same local carrier replica or local carrier replicas that only differ by an integer number of cycles. Under this assumption, the terms $\gamma_A^i - \gamma_A^0$ and $\gamma_B^i - \gamma_B^0$ are both integers and, thus, their difference is an integer.

This integer ambiguity is also constant provided that the phase-lock loops (PLLs) in both receivers for both satellites do not slip cycles. If any of these four carrier-phases drop or gain any cycles, then the integer ambiguity will no longer be the same and the CDGPS solution will suffer. For satellites above 10 or 15 degrees in elevation, cycle slips are rare if there are no obstructions blocking the line-of-sight signal. However, cycle slip robustness is still an important issue for both receiver design and CDGPS algorithm design.

The only remaining error source in the double-differenced measurements, besides noise, is the double-differenced multipath error. The worst-case carrier-phase multipath error is only on the order of centimeters, while the pseudorange multipath error can be as high as 20 m. This means that multipath will not significantly degrade performance of CDGPS once the carrier-phase integer ambiguities have been determined, since the pseudorange measurements have almost no effect on the pose solution at this point. However, pseudorange multipath errors can cause difficulty during the initial phase when the integer ambiguities are being determined. Multipath errors are also highly correlated in time, which further complicates the issue. Additionally, carrier-phase multipath may cause cycle slips, which cuts against robustness of the system. Multipath errors can largely be removed by masking out low elevation satellites, but any tall structures in the area of operation may create multipath reflections. In the end, the integer ambiguities will converge to the correct value, but it will take significantly longer and the carrier-phase may slip cycles in the presence of severe multipath.

3.2.1.4 Linearized Double-Differenced pseudorange and Carrier-Phase Measurements Models

Eqs. 3.18 and 3.19 are linearized about the a priori estimate of the real-valued portion of the state assuming that multipath errors are not present. The resulting linearized double-differenced measurements are

$$\begin{aligned}
z_{\rho}^{i0}(k) &= \nabla\Delta\rho_{AB}^{i0}(k) - \nabla\Delta\bar{r}_{AB}^{i0}(k) \\
&= \left(\hat{r}_{ECEF}^{0,B}(k) - \hat{r}_{ECEF}^{i,B}(k)\right)^{T} \delta\mathbf{x}_{ECEF}^{B}(k) \\
&+ 2\left(\hat{r}_{ECEF}^{0,B}(k) - \hat{r}_{ECEF}^{i,B}(k)\right)^{T} \left[\left(R\left(\bar{\mathbf{q}}_{ECEF}^{B}(k)\right)\mathbf{x}_{B}^{GPS}\right)\times\right] \delta\mathbf{e}_{ECEF}^{B}(k) \\
&+ \nabla\Delta w_{\rho,AB}^{i0}(k)
\end{aligned} \tag{3.25}$$

$$\begin{aligned}
z_{\phi}^{i0}(k) &= \lambda_{L1}\nabla\Delta\phi_{AB}^{i0}(k) - \nabla\Delta\bar{r}_{AB}^{i0}(k) \\
&= \left(\hat{r}_{ECEF}^{0,B}(k) - \hat{r}_{ECEF}^{i,B}(k)\right)^{T} \delta\mathbf{x}_{ECEF}^{B}(k) \\
&+ 2\left(\hat{r}_{ECEF}^{0,B}(k) - \hat{r}_{ECEF}^{i,B}(k)\right)^{T} \left[\left(R\left(\bar{\mathbf{q}}_{ECEF}^{B}(k)\right)\mathbf{x}_{B}^{GPS}\right)\times\right] \delta\mathbf{e}_{ECEF}^{B}(k) \\
&+ \lambda_{L1}N_{AB}^{i0} + \nabla\Delta w_{\phi,AB}^{i0}(k)
\end{aligned} \tag{3.26}$$

where $\nabla\Delta\bar{r}^{i0}_{AB}(k)$ is the expected double-differenced range based on satellite ephemeris and the a priori state estimate, $\hat{r}^{i,B}_{ECEF}(k)$ is the unit vector pointing to the ith SV from RX B, $\delta\mathbf{x}^{B}_{ECEF}(k)$ is the a prosteriori correction to the position estimate, $[(\cdot)\times]$ is the cross-product equivalent matrix of the argument, and $\delta\mathbf{e}^{B}_{ECEF}(k)$ is the minimal representation of the differential quaternion, defined in Appendix A, representing the a posteriori correction to the attitude estimate.

If both receivers are tracking the same M+1 satellites, then M linearized double-differenced measurements are obtained of the form given in Eqs. 3.25 and 3.26. Gathering these M equations into matrix form gives

$$\begin{bmatrix} \mathbf{z}_{\rho}(k) \\ \mathbf{z}_{\phi}(k) \end{bmatrix} = \begin{bmatrix} H_{\rho,x}(k) & 0 \\ H_{\phi,x}(k) & H_{\phi,N} \end{bmatrix} \begin{bmatrix} \delta\mathbf{x}(k) \\ \mathbf{N} \end{bmatrix} + \begin{bmatrix} \nabla\Delta\mathbf{w}_{\rho} \\ \nabla\Delta\mathbf{w}_{\phi} \end{bmatrix} \tag{3.27}$$

where $\delta\mathbf{x}(k)$ is the a posteriori correction to the real-valued component of the state and

$$\begin{aligned} H_{\rho,x}(k) &= H_{\phi,x}(k) \\ &= \begin{bmatrix} \left.\frac{\partial\nabla\Delta\rho^{10}_{AB}}{\partial\mathbf{x}^{B}_{ECEF}}\right|_{\bar{\mathbf{X}}(k)} & 0_{1\times 6} & \left.\frac{\partial\nabla\Delta\rho^{10}_{AB}}{\partial\delta\mathbf{e}^{B}_{ECEF}}\right|_{\bar{\mathbf{X}}(k)} & 0_{1\times 7} \\ \vdots & \vdots & \vdots & \vdots \\ \left.\frac{\partial\nabla\Delta\rho^{M0}_{AB}}{\partial\mathbf{x}^{B}_{ECEF}}\right|_{\bar{\mathbf{X}}(k)} & 0_{1\times 6} & \left.\frac{\partial\nabla\Delta\rho^{M0}_{AB}}{\partial\delta\mathbf{e}^{B}_{ECEF}}\right|_{\bar{\mathbf{X}}(k)} & 0_{1\times 7} \end{bmatrix} \end{aligned} \tag{3.28}$$

$$H_{\phi,N} = \lambda_{L1} I \tag{3.29}$$

where I is the identity matrix. The partial derivatives in Eq. 3.28 can be determined from Eq. 3.25 as

$$\left.\frac{\partial\nabla\Delta\rho^{i0}_{AB}}{\partial\mathbf{x}^{B}_{ECEF}}\right|_{\bar{\mathbf{X}}(k)} = \Big(\hat{r}^{0,B}_{ECEF}(k) - \hat{r}^{i,B}_{ECEF}(k)\Big)^{T} \tag{3.30}$$

$$\left.\frac{\partial \nabla\Delta\rho^{i0}_{AB}}{\partial \delta \mathbf{e}^{B}_{ECEF}}\right|_{\bar{\mathbf{X}}(k)} = 2\left(\hat{r}^{0,B}_{ECEF}(k) - \hat{r}^{i,B}_{ECEF}(k)\right)^{T}\left[\left(R\left(\bar{\mathbf{q}}^{B}_{ECEF}(k)\right)\mathbf{x}^{GPS}_{B}\right)\times\right] \tag{3.31}$$

The covariance matrices for the double-differenced measurement noise can be assembled from Eqs. 3.21, 3.22, 3.23, and 3.24 as

$$\begin{aligned} R_{\rho}(k) &= E\left[\nabla\Delta\mathbf{w}_{\rho}(k)\nabla\Delta\mathbf{w}^{T}_{\rho}(k)\right] \\ &= \begin{bmatrix} \left(\sigma^{10}_{\rho,AB}(k)\right)^{2} & \left(\sigma^{0}_{\rho,AB}(k)\right)^{2} & \cdots & \left(\sigma^{0}_{\rho,AB}(k)\right)^{2} \\ \left(\sigma^{0}_{\rho,AB}(k)\right)^{2} & \left(\sigma^{20}_{\rho,AB}(k)\right)^{2} & \cdots & \left(\sigma^{0}_{\rho,AB}(k)\right)^{2} \\ \vdots & & \ddots & \vdots \\ \left(\sigma^{0}_{\rho,AB}(k)\right)^{2} & & \cdots & \left(\sigma^{M0}_{\rho,AB}(k)\right)^{2} \end{bmatrix} \end{aligned} \tag{3.32}$$

$$\begin{aligned} R_{\phi}(k) &= E\left[\nabla\Delta\mathbf{w}_{\phi}(k)\nabla\Delta\mathbf{w}^{T}_{\phi}(k)\right] \\ &= \begin{bmatrix} \left(\sigma^{10}_{\phi,AB}(k)\right)^{2} & \left(\sigma^{0}_{\phi,AB}(k)\right)^{2} & \cdots & \left(\sigma^{0}_{\phi,AB}(k)\right)^{2} \\ \left(\sigma^{0}_{\phi,AB}(k)\right)^{2} & \left(\sigma^{20}_{\phi,AB}(k)\right)^{2} & \cdots & \left(\sigma^{0}_{\phi,AB}(k)\right)^{2} \\ \vdots & & \ddots & \vdots \\ \left(\sigma^{0}_{\phi,AB}(k)\right)^{2} & & \cdots & \left(\sigma^{M0}_{\phi,AB}(k)\right)^{2} \end{bmatrix} \end{aligned} \tag{3.33}$$

3.2.2 INS Measurement Models

An INS is typically composed of an IMU with a three-axis accelerometer, a three-axis gyro, and a magnetometer. The accelerometer measurements are useful for propagating position forward in time and estimation of the gravity vector. Estimation of the gravity vector can only be performed using a low-pass filter of the accelerometer measurements under the assumption that the IMU is not subject to long-term sustained accelerations. This is typically the case for pedestrian and vehicular motion over time constants of a minute or longer. The magnetometer can also be used to estimate the direction of magnetic north under the assumption that magnetic disturbances are negligible or calibrated out of the system. However, a low-pass filter with a large time constant must also be applied to the magnetometer measurements to accurately estimate the direction of magnetic north, since the Earth's magnetic field is extremely weak.

Once the gravity vector and direction of magnetic north have been determined, the IMU is capable of estimating its attitude relative to the local ENU frame after correcting for magnetic declination. Due to the long time constant filters, the attitude estimate must be propagated using the angular velocity measurements from the gyro to provide accurate attitude during dynamics. This means that the attitude estimated by the IMU is highly correlated with the angular velocity measurements.

The navigation filter presented in this thesis relies on the accelerometer measurements and attitude estimates from the IMU. The accelerometer measurements aid in propagating the state forward in time, while the IMU estimated attitude provides the primary sense of absolute attitude for the system. As demonstrated in Sec. 2.1, coupled GPS and visual SLAM is capable of estimating absolute attitude, but this navigation filter has difficulty doing so without an IMU because of the need to additionally estimate the similarity transform between ECEF and the V-frame. Therefore, the navigation filter must rely on the IMU estimated attitude. Since the angular velocity measurements are highly correlated with the IMU estimated attitude, the angular velocity measurements are discarded.

3.2.2.1 Accelerometer Measurement Model

The accelerometer measurements from the IMU are modeled as follows

$$
\begin{aligned}
\mathbf{f}(k) = {} & R\left(\mathbf{q}_{ECEF}^{B}(k)\right)^{T}\left(\dot{\mathbf{v}}_{ECEF}^{B}(k) + 2\left[\boldsymbol{\omega}_{E}\times\right]\mathbf{v}_{ECEF}^{B}(k)\right) \\
& + R\left(\mathbf{q}_{B}^{ENU}(k)\right)\begin{bmatrix} 0 \\ 0 \\ g(k) \end{bmatrix} + \mathbf{b}_{a}(k) + \boldsymbol{\nu}_{a}'(k)
\end{aligned}
\tag{3.34}
$$

where $\mathbf{f}(k)$ is the accelerometer measurement, $\boldsymbol{\omega}_E$ is the angular velocity vector of the Earth, $\boldsymbol{\nu}_a'(k)$ is zero-mean Gaussian white noise with a diagonal covariance matrix, and $g(k)$ is the gravitational acceleration of Earth at the position of the IMU that is approximated as

$$
g(k) = \frac{G_E}{\left|\left|\mathbf{x}_{ECEF}^{B}(k)\right|\right|^{2}}
\tag{3.35}
$$

where G_E is the gravitational constant of Earth. This accelerometer measurement model is similar to the model in [53]. Equation 3.34 can be solved for the acceleration of the IMU expressed in ECEF to obtain

$$
\begin{aligned}
\dot{\mathbf{v}}_{ECEF}^{B}(k) = {} & R\left(\mathbf{q}_{ECEF}^{B}(k)\right)\left(\mathbf{f}(k) - \mathbf{b}_{a}(k)\right) + R(\mathbf{q}_{ECEF}^{ENU}(k))\begin{bmatrix} 0 \\ 0 \\ -g(k) \end{bmatrix} \\
& - 2\left[\boldsymbol{\omega}_{E}\times\right]\mathbf{v}_{ECEF}^{B}(k) + \boldsymbol{\nu}_{a}(k)
\end{aligned}
\tag{3.36}
$$

where $\boldsymbol{\nu}_a(k)$ is a rotated version of $\boldsymbol{\nu}_a'(k)$ and thus identically distributed. These measurements will be used in the dynamics model in Sec. 3.3.1.

3.2.2.2 INS Attitude Estimate Model

The attitude estimates from the IMU are modeled as follows

$$
\begin{aligned}
\tilde{\mathbf{q}}_{ENU}^{B}(k) &= \mathbf{q}_{ENU}^{ECEF}(k) \otimes \mathbf{q}_{ECEF}^{B}(k) + \mathbf{w}_{q}^{I\prime}(k) \\
&= \mathbf{q}_{ENU}^{ECEF}(k) \otimes \delta\mathbf{q}_{ECEF}^{B}(k) \otimes \bar{\mathbf{q}}_{ECEF}^{B}(k) + \mathbf{w}_{q}^{I\prime}(k)
\end{aligned}
\tag{3.37}
$$

where $\tilde{\mathbf{q}}^B_{ENU}(k)$ is the IMU attitude estimate and $\mathbf{w}^{I\prime}_q(k)$ is zero-mean Gaussian white noise with a diagonal covariance matrix. Modeling the noise on the attitude estimates as white is not strictly correct as there will be strongly time-correlated biases in the attitude estimates from the IMU, but these time-correlated errors are assumed small. The quaternion $\mathbf{q}^{ECEF}_{ENU}(k)$ can be computed from the a priori estimate of the position of the IMU. This dependence on position, however, will be ignored for linearization, since it is extremely weak. In linearizing Eq. 3.37, the following relation is defined based on the quaternion left ($[\cdot]$) and right ($\{\cdot\}$) multiplication matrices defined in Appendix A

$$\begin{bmatrix} H^I_{q_0,\delta(q_0)^B_{ECEF}}(k) & H^I_{q_0,\delta\mathbf{e}^B_{ECEF}}(k) \\ H^I_{e,\delta(q_0)^B_{ECEF}}(k) & H^I_{e,\delta\mathbf{e}^B_{ECEF}}(k) \end{bmatrix} = \left[\mathbf{q}^{ECEF}_{ENU}(k)\right]\left\{\bar{\mathbf{q}}^B_{ECEF}(k)\right\} \tag{3.38}$$

The linearized attitude measurement can then be expressed in minimal form as

$$\begin{aligned} \mathbf{z}^I_q(k) &= \tilde{\mathbf{e}}^B_{ENU}(k) - \bar{\mathbf{e}}^B_{ENU}(k) \\ &= \begin{bmatrix} H^I_{q,x} & 0 \end{bmatrix} \begin{bmatrix} \delta\mathbf{x} \\ \mathbf{N} \end{bmatrix} + \mathbf{w}^I_q(k) \end{aligned} \tag{3.39}$$

where $\tilde{\mathbf{e}}^B_{ENU}(k)$ and $\bar{\mathbf{e}}^B_{ENU}(k)$ are the measured and expected values of the vector portion of the quaternion $\mathbf{q}^B_{ENU}(k)$ respectively, $\mathbf{w}^I_q(k)$ is the last three elements of $\mathbf{w}^{I\prime}_q(k)$, and

$$H^I_{e,x}(k) = \begin{bmatrix} 0_{3\times9} & H^I_{q,\delta\mathbf{e}^B_{ECEF}}(k) & 0_{3\times7} \end{bmatrix} \tag{3.40}$$

The covariance matrix for these attitude estimates is

$$R^I_q = \left(\sigma^I_q\right)^2 I \tag{3.41}$$

A reasonable value for σ_q^I is 0.01, which corresponds to an attitude error of approximately 2°. Since the IMU considered here includes a magnetometer, the IMU's estimate of attitude does not drift.

3.2.3 VNS Measurement Models

A BA-based stand-alone visual SLAM algorithm is employed to provide relative pose estimates of the system [45]. These estimates are represented in the V-frame, which has an unknown translation, orientation, and scale-factor relative to ECEF that must be estimated. The visual SLAM algorithm does not provide covariances for its relative pose estimates to reduce computational expense of the algorithm. Therefore, all noises for the visual SLAM estimates are assumed to be independent. Although this is not strictly true, it is not an unreasonable approximation.

3.2.3.1 VNS Position Estimate Model

The position estimates from the visual SLAM algorithm are modeled as

$$\tilde{\mathbf{x}}_V^C(k) = \lambda R\left(\mathbf{q}_V^{ECEF}\right)\left(\mathbf{x}_{ECEF}^B(k) + R\left(\mathbf{q}_{ECEF}^B(k)\right)\mathbf{x}_B^C - \mathbf{x}_{ECEF}^V\right) + \mathbf{w}_p^V(k) \tag{3.42}$$

where $\tilde{\mathbf{x}}_V^C(k)$ is the position estimate of the camera in the V-frame, $\mathbf{x}_B^C$ is the position of the camera lens in the B-frame, and $\mathbf{w}_p^V(k)$ is zero-mean Gaussian white noise with a diagonal covariance matrix given by

$$R_p^V = \left(\sigma_p^V\right)^2 I \tag{3.43}$$

The value of σ_p^V depends heavily on the depth of the scene features tracked by the visual SLAM algorithm. A reasonable value of σ_p^V for a depth of a few meters is 1 cm.

The measurement model from Eq. 3.42 is linearized about the a priori state estimate to obtain

$$\begin{aligned} z_p^V(k) &= \tilde{\mathbf{x}}_V^C(k) - \bar{\lambda}(k) R\left(\bar{\mathbf{q}}_V^{ECEF}(k)\right)\left(\bar{\mathbf{x}}_{ECEF}^B(k) + R\left(\bar{\mathbf{q}}_{ECEF}^B(k)\right)\mathbf{x}_B^C - \bar{\mathbf{x}}_{ECEF}^V(k)\right) \\ &= \begin{bmatrix} H_{p,x}^V & 0 \end{bmatrix} \begin{bmatrix} \delta\mathbf{x} \\ \mathbf{N} \end{bmatrix} + \mathbf{w}_p^V(k) \end{aligned} \tag{3.44}$$

where

$$\begin{aligned} &H_{p,x}^V(k) \\ &= \begin{bmatrix} \left(\bar{\lambda}(k) R\left(\bar{\mathbf{q}}_V^{ECEF}(k)\right)\right)^T \\ 0_{6\times 3} \\ \left(2\bar{\lambda}(k) R\left(\bar{\mathbf{q}}_V^{ECEF}(k)\right)\left[\left(R\left(\bar{\mathbf{q}}_{ECEF}^B(k)\right)\mathbf{x}_B^C\right)\times\right]\right)^T \\ \left(-\bar{\lambda}(k) R\left(\bar{\mathbf{q}}_V^{ECEF}(k)\right)\right)^T \\ \left(2\bar{\lambda}(k)\left[\left(R\left(\bar{\mathbf{q}}_V^{ECEF}(k)\right)\left(\bar{\mathbf{x}}_{ECEF}^B(k) + R\left(\bar{\mathbf{q}}_{ECEF}^B(k)\right)\mathbf{x}_B^C - \bar{\mathbf{x}}_{ECEF}^V(k)\right)\right)\times\right]\right)^T \\ \left(R\left(\bar{\mathbf{q}}_V^{ECEF}(k)\right)\left(\bar{\mathbf{x}}_{ECEF}^B(k) + R\left(\bar{\mathbf{q}}_{ECEF}^B(k)\right)\mathbf{x}_B^C - \bar{\mathbf{x}}_{ECEF}^V(k)\right)\right)^T \end{bmatrix}^T \end{aligned} \tag{3.45}$$

3.2.3.2 VNS Attitude Estimate Model

The attitude estimates from the visual SLAM algorithm are modeled as

$$\begin{aligned} \tilde{\mathbf{q}}_V^C(k) &= \mathbf{q}_V^{ECEF} \otimes \mathbf{q}_{ECEF}^B(k) \otimes \mathbf{q}_B^C + \mathbf{w}_q^{V}{'}(k) \\ &= \delta\mathbf{q}_V^{ECEF}(k) \otimes \bar{\mathbf{q}}_V^{ECEF}(k) \otimes \delta\mathbf{q}_{ECEF}^B(k) \otimes \bar{\mathbf{q}}_{ECEF}^B(k) \otimes \mathbf{q}_B^C + \mathbf{w}_q^{V}{'}(k) \end{aligned} \tag{3.46}$$

where $\tilde{\mathbf{q}}_V^C(k)$ is the attitude estimate of the camera relative to the V-frame, $\mathbf{q}_B^C$ is the attitude of the camera relative to the B-frame, and $\mathbf{w}_q^{V}{'}(k)$ is zero-mean Gaussian white noise with a diagonal covariance matrix. In linearizing Eq. 3.46, the following relations are defined based on the quaternion left and right multiplication matrices defined in Appendix A

$$\left[\begin{array}{cc} H^V_{q_0,\delta(q_0)^B_{ECEF}}(k) & H^V_{q_0,\delta\mathbf{e}^B_{ECEF}}(k) \\ H^V_{e,\delta(q_0)^B_{ECEF}}(k) & H^V_{e,\delta\mathbf{e}^B_{ECEF}}(k) \end{array} \right] = \left[\bar{\mathbf{q}}^{ECEF}_V(k)\right] \left\{\bar{\mathbf{q}}^B_{ECEF}(k)\right\} \left\{\mathbf{q}^C_B(k)\right\} \tag{3.47}$$

$$\left[\begin{array}{cc} H^V_{q_0,\delta(q_0)^{ECEF}_V}(k) & H^V_{q_0,\delta\mathbf{e}^{ECEF}_V}(k) \\ H^V_{e,\delta(q_0)^{ECEF}_V}(k) & H^V_{e,\delta\mathbf{e}^{ECEF}_V}(k) \end{array} \right] = \left\{\bar{\mathbf{q}}^{ECEF}_V(k)\right\} \left\{\bar{\mathbf{q}}^B_{ECEF}(k)\right\} \left\{\mathbf{q}^C_B(k)\right\} \tag{3.48}$$

The linearized attitude measurement can then be expressed in minimal form as

$$\begin{aligned} \mathbf{z}^V_q(k) &= \tilde{\mathbf{e}}^C_V(k) - \bar{\mathbf{e}}^C_V(k) \\ &= \left[\begin{array}{cc} H^V_{q,x} & 0 \end{array} \right] \left[\begin{array}{c} \delta\mathbf{x} \\ \mathbf{N} \end{array} \right] + \mathbf{w}^V_q(k) \end{aligned} \tag{3.49}$$

where $\tilde{\mathbf{e}}^C_V(k)$ and $\bar{\mathbf{e}}^C_V(k)$ are the measured and expected values of the vector portion of the quaternion $\mathbf{q}^C_V(k)$ respectively, $\mathbf{w}^V_q(k)$ is the last three elements of $\mathbf{w}^{V\prime}_q(k)$, and

$$H^V_{q,x}(k) = \left[\begin{array}{ccccc} 0_{3\times 9} & H^V_{e,\delta\mathbf{e}^B_{ECEF}}(k) & 0_{3\times 3} & H^V_{e,\delta\mathbf{e}^{ECEF}_V}(k) & 0_{3\times 1} \end{array} \right] \tag{3.50}$$

The covariance matrix for these attitude estimates is

$$R^V_q = \left(\sigma^V_q\right)^2 I \tag{3.51}$$

A reasonable value for σ^V_q is 0.005, which corresponds to an attitude error of approximately 1°.

3.3 Dynamics Models

The development of the dynamics models presented in this section follows the notation presented in Appendix B for the propagation step. Two separate dynamics models are used in the navigation filter depending on whether or not INS measurements are provided to the filter.

3.3.1 INS Dynamics Model

Whenever INS measurements are present, the navigation filter uses the accelerometer measurements from the IMU to propagate the position and velocity of the system forward in time using Eq. 3.36. The accelerometer bias is modeled as a first-order Gauss-Markov process. Angular velocity measurements from the IMU cannot be used for propagation of the attitude of the system since the filter uses attitude estimates from the IMU, which are highly correlated with the angular velocity measurements. Therefore, the attitude is held constant over the propagation step with some added process noise to account for the unmodeled angular velocity. All other parameters in the real-valued portion of the state are constants and are modeled as such. The integer ambiguities are excluded from the propagation step, since they are constants anyways. However, the cross-covariance between the real-valued portion of the state and the integer ambiguities is propagated forward properly. This is explained in greater detail in Chapter 4.

The resulting dynamics model for the state is

$$f\left(\mathbf{x}(t), \mathbf{u}(t), t\right) = \begin{bmatrix} \mathbf{v}^{B}_{ECEF}(t) \\ \left(\begin{matrix} R\left(\mathbf{q}^{B}_{ECEF}(t)\right)\left(\mathbf{f}(t) - \mathbf{b}_a(t)\right) + R\left(\mathbf{q}^{ENU}_{ECEF}(t)\right)\begin{bmatrix} 0 \\ 0 \\ -g(t) \end{bmatrix} \\ -2\left[\boldsymbol{\omega}_E \times\right]\mathbf{v}^{B}_{ECEF}(t) \end{matrix} \right) \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{3.52}$$

where $\mathbf{u}(t)$ is the input vector given by

$$\mathbf{u}(t) = \begin{bmatrix} \mathbf{f}(t) \\ -g(t) \end{bmatrix} \tag{3.53}$$

Process noise is added to the dynamics model to account for un-modeled effects and is given by

$$D(t)\mathbf{v}'(t) = \begin{bmatrix} 0_{3\times 9} \\ I_{9\times 9} \\ 0_{7\times 9} \end{bmatrix} \begin{bmatrix} \boldsymbol{\nu}_a(t) \\ \boldsymbol{\nu}_b(t) \\ \boldsymbol{\nu}_\omega(t) \end{bmatrix} \tag{3.54}$$

The process noise covariance is

$$Q'(t) = E\left[\mathbf{v}'(t)\mathbf{v}'^{T}(t)\right] = \begin{bmatrix} \sigma_a^2 I & 0 & 0 \\ 0 & \sigma_b^2 I & 0 \\ 0 & 0 & \frac{1}{4}\sigma_\omega^2 I \end{bmatrix} \tag{3.55}$$

The term $\frac{1}{4}\sigma_\omega^2$ comes from the following relation which can be derived from quaternion kinematics, presented in Appendix A, under the initial condition that $\delta\mathbf{e}^{B}_{ECEF} = 0$

$$\delta\dot{\mathbf{e}}^{B}_{ECEF}(t) = \frac{1}{2}\boldsymbol{\omega}(t) = \boldsymbol{\nu}_\omega(t) \tag{3.56}$$

where $\boldsymbol{\omega}(t)$ is the angular velocity vector of the system which is modeled as zero-mean Gaussian white noise with a diagonal covariance matrix. The values of σ_a and σ_b from Eq. 3.55 depend on the quality of the IMU and can typically be found on the IMU's specifications provided by the manufacturer. On the other hand, σ_ω depends on the expected dynamics of the system.

Since the IMU measurements are reported at a rate of 100 Hz, the propagation interval, Δt, is at most 10 ms. This interval is small enough that the dynamics model can be assumed constant over the interval and higher order terms in Δt are negligible compared to lower order terms.

Under this assumption, the dynamics model is then integrated over the propagation interval to form a difference equation of the form

$$\mathbf{x}(k+1) \approx \mathbf{x}(k) + \Delta t f\left(\mathbf{x}(k), \mathbf{u}(k), t_k\right) + \Gamma(k)\mathbf{v}(k) \tag{3.57}$$

where $\mathbf{v}(k)$ is the discrete-time zero-mean Gaussian white process noise vector, and

$$\Gamma(k) = \begin{bmatrix} I_{12\times12} \\ 0_{7\times12} \end{bmatrix} \tag{3.58}$$

The partial derivative of the difference equation from Eq. 3.57 is taken with respect to the state and evaluated at the a posteriori state estimate at time t_k to obtain the state transition matrix

$$\begin{aligned} F(k) &= I + \Delta t \\ &\times \begin{bmatrix} 0_{3\times3} & I_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times7} \\ 0_{3\times3} & -2\left[\boldsymbol{\omega}_E\times\right] & -R\left(\hat{\mathbf{q}}^B_{ECEF}(k)\right) & 2\left[\left(R\left(\hat{\mathbf{q}}^B_{ECEF}(k)\right)\left(\mathbf{f}(k)-\hat{\mathbf{b}}_a(k)\right)\right)\times\right] & 0_{3\times7} \\ 0_{13\times3} & 0_{13\times3} & 0_{13\times3} & 0_{13\times3} & 0_{13\times7} \end{bmatrix} \end{aligned} \tag{3.59}$$

This linearization neglects the extremely weak coupling of the position of the system to the terms $R\left(\mathbf{q}_{ECEF}^{ENU}(k)\right)$ and $g(k)$. The discrete-time process noise covariance, $Q(k)$, can be determined as described in Appendix B. This covariance matrix is given by

$$Q(k) = E\left[\mathbf{v}(k)\mathbf{v}^T(k)\right] = \begin{bmatrix} Q_{(1,1)}(k) & Q_{(1,2)}(k) & 0 & 0 \\ Q_{(1,2)}^T(k) & Q_{(2,2)}(k) & Q_{(2,3)}(k) & Q_{(2,4)}(k) \\ 0 & Q_{(2,3)}^T(k) & Q_{(3,3)}(k) & 0 \\ 0 & Q_{(2,4)}^T(k) & 0 & Q_{(4,4)}(k) \end{bmatrix} \tag{3.60}$$

where the terms in $Q(k)$ are as follows

$$Q_{(1,1)}(k) = \frac{1}{3}\Delta t^3 \sigma_a^2 I \tag{3.61}$$

$$Q_{(1,2)}(k) = \frac{1}{2}\Delta t^2 \sigma_a^2 I \tag{3.62}$$

$$\begin{aligned} Q_{(2,2)}(k) = {} & \left(\Delta t \sigma_a^2 + \frac{1}{3}\Delta t^3 \sigma_b^2\right) I + \frac{1}{3}\Delta t^3 \sigma_\omega^2 \\ & \times \left[\left(R\left(\hat{\mathbf{q}}_{ECEF}^B(k)\right)\left(\mathbf{f}(k) - \hat{\mathbf{b}}_a(k)\right)\right)\times\right]\left[\left(R\left(\hat{\mathbf{q}}_{ECEF}^B(k)\right)\left(\mathbf{f}(k) - \hat{\mathbf{b}}_a(k)\right)\right)\times\right]^T \\ & \approx \Delta t \sigma_a^2 I \end{aligned} \tag{3.63}$$

$$Q_{(2,3)}(k) = -\frac{1}{2}\Delta t^2 \sigma_b^2 R\left(\hat{\mathbf{q}}_{ECEF}^B(k)\right) \tag{3.64}$$

$$Q_{(2,4)}(k) = \frac{1}{4}\Delta t^2 \sigma_\omega^2 \left[\left(R\left(\hat{\mathbf{q}}_{ECEF}^B(k)\right)\left(\mathbf{f}(k) - \hat{\mathbf{b}}_a(k)\right)\right)\times\right] \tag{3.65}$$

$$Q_{(3,3)}(k) = \Delta t \sigma_b^2 I \tag{3.66}$$

$$Q_{(4,4)}(k) = \frac{1}{4} \Delta t \sigma_\omega^2 I \tag{3.67}$$

3.3.2 INS-Free Dynamics Model

Whenever INS measurements are not present, the INS-free dynamics model reverts to a velocity-random-walk model for the velocity in place of the accelerometer measurements. This is necessary because no other information about the dynamics of the system is available. All other states are propagated using models identical to those for the INS dynamics model. The accelerometer bias would typically not be represented in this model because this model would only be used if there were no accelerometer measurements and thus no need to have the bias in the state vector. However, it is maintained here primarily for notational consistency. The filter could also revert to this model if the accelerometer measurements were temporarily lost for whatever reason and it was desirable to maintain the accelerometer bias in the state.

The resulting dynamics model for the state is simply

$$f\left(\mathbf{x}(t), \mathbf{u}(t), t\right) = \begin{bmatrix} \mathbf{v}_{ECEF}^{B}(t) \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{3.68}$$

with additive process noise given by

$$D(t)\mathbf{v}'(t) = \begin{bmatrix} 0_{3\times9} \\ I_{9\times9} \\ 0_{7\times9} \end{bmatrix} \begin{bmatrix} \boldsymbol{\nu}_{\dot{v}}(t) \\ \boldsymbol{\nu}_{b}(t) \\ \boldsymbol{\nu}_{\omega}(t) \end{bmatrix} \tag{3.69}$$

The process noise covariance is assumed to be

$$Q'(t) = E\left[\mathbf{v}'(t)\mathbf{v}'^{T}(t)\right] = \begin{bmatrix} \sigma_{\dot{v}}^2 I & 0 & 0 \\ 0 & \sigma_b^2 I & 0 \\ 0 & 0 & \frac{1}{4}\sigma_\omega^2 I \end{bmatrix} \tag{3.70}$$

where $\sigma_{\dot{v}}$ and σ_ω depend on the expected dynamics of the system and σ_b can be obtained from the IMU's specifications.

These propagation steps occur much less often than with the INS dynamics model. For a CDGPS-only filter, the propagation interval could be as large as 1 s, since many receivers only report observables at 1 s intervals. Therefore, the assumptions about the interval being small that were made for the INS dynamics model cannot be made here. However, this dynamics model is in fact linear and can be integrated directly to obtain the difference equation

$$\mathbf{x}(k+1) = F(k)\mathbf{x}(k) + \Gamma(k)\mathbf{v}(k) \tag{3.71}$$

where $\Gamma(k)$ is the same as in Eq. 3.58. It can easily be shown that the state transition matrix and discrete-time process noise covariance for this dynamics model are

$$F(k) = \begin{bmatrix} I_{3\times3} & \Delta t I_{3\times3} & 0_{3\times13} \\ 0_{3\times3} & I_{3\times3} & 0_{3\times13} \\ 0_{13\times3} & 0_{13\times3} & I_{13\times13} \end{bmatrix} \tag{3.72}$$

and

$$Q(k) = E\left[\mathbf{v}(k)\mathbf{v}^T(k)\right] = \begin{bmatrix} \frac{1}{3}\Delta t^3 \sigma_{\dot{v}}^2 I & \frac{1}{2}\Delta t^2 \sigma_{\dot{v}}^2 I & 0 & 0 \\ \frac{1}{2}\Delta t^2 \sigma_{\dot{v}}^2 I & \Delta t \sigma_{\dot{v}}^2 I & 0 & 0 \\ 0 & 0 & \Delta t \sigma_{\dot{b}}^2 I & 0 \\ 0 & 0 & 0 & \frac{1}{4}\Delta t \sigma_{\omega}^2 I \end{bmatrix} \tag{3.73}$$

Chapter 4

Navigation Filter

Measurement and dynamics models for a mobile AR system employing double-differenced GPS observables measurements, IMU accelerometer measurements and attitude estimates, and relative pose estimates from a stand-alone visual SLAM algorithm were derived in Chapter 3. With these measurement and dynamics models, a navigation filter for the AR system is designed that couples CDGPS, visual SLAM, and an INS. This navigation filter is capable of providing at least centimeter-level position and degree-level attitude accuracy in open outdoor areas. If the visual SLAM algorithm was coupled tighter to the GPS and INS measurements, then this system could also transition indoors and maintain highly-accurate global pose for a limited time without GPS availability. The current filter only operates in post-processing, but could be made to run in real time.

This chapter presents a square-root EKF (SREKF) implementation of such a navigation filter. The chapter begins by discussing how the filter state is encoded as measurement equations while accomodating the use of quaternions and a mixed real-integer valued state. Then, the measurement update and propagation steps are outlined. The method for handling changes in the satellites tracked by the GPS receivers is also discussed. A general overview of the SREKF is provided in Appendix B for further reference.

4.1 Filter State Encoded as Measurements

In square-root filter implementations, the state estimate and state covariance are represented by a set of measurement equations. These measurement equations express the filter state as a measurement of the true state with added zero-mean Gaussian white noise that has a covariance matrix equal to the state covariance. After normalizing these measurements so that the noise has a covariance matrix of identity, the state measurement equations are given by

$$\mathbf{z}_X(k) = R_{XX}(k)\mathbf{X}(k) + \mathbf{w}_X(k) \tag{4.1}$$

where $\mathbf{z}_X(k)$ are the state measurements, $R_{XX}(k)$ is the upper-triangular Cholesky factorization of the inverse of the state covariance $P^{-1}(k)$, and $\mathbf{w}_X(k)$ is the normalized zero-mean Gaussian white noise.

For the filter reported in this thesis, these equations are expressed slightly differently to properly handle the integer portion of the state and the elements of the state which are quaternion attitude representations. To handle the integer portion of the state, the state is simply partitioned into real-valued and integer components as mentioned in Sec. 3.1.3. This partitioning is useful in solving for the state after measurement update and propagation steps, which is described in Sec. 4.1.1. To handle the quaternions properly, the filter must ensure that the quaternions are constrained to have unity magnitude, as required by the definition of a quaternion, during measurement update and propagation steps. This constraint is enforced by expressing the quaternions in the state instead as differential quaternions, which can be reduced to a minimal attitude representation that does not require the unity magnitude constraint through a small angle assumption [48], as shown in Appendix A. These differential quaternions represent a small rotation from the current best estimate of the corresponding quaternion as seen in Eq. 3.6.

Based on these considerations, the resulting state measurement equations are

$$\begin{bmatrix} \mathbf{z}_x(k) \\ \mathbf{z}_N(k) \end{bmatrix} = \begin{bmatrix} R_{xx}(k) & R_{xN}(k) \\ 0 & R_{NN}(k) \end{bmatrix} \begin{bmatrix} \mathbf{x}(k) \\ \mathbf{N} \end{bmatrix} + \begin{bmatrix} \mathbf{w}_x(k) \\ \mathbf{w}_N(k) \end{bmatrix} \tag{4.2}$$

where the quaternion elements of $\mathbf{x}(k)$ are stored separately and replaced by differential quaternions in minimal form. This set of equations is used in the filter in place of Eq. 4.1, which is used in the standard SREKF shown in Appendix B.

4.1.1 Computing the State and State Covariance from the State Measurement Equations

Equation 4.2 is updated in the filter as new measurements are collected through a measurement update step and as the filter propagates the state forward in time through a propagation step. Whenever the state estimate and state covariance are desired, they can be computed from Eq. 4.2 as follows

1. The integer valued portion of the state is first determined through an integer least squares (ILS) solution algorithm taking $\mathbf{z}_N(k)$ and $R_{NN}(k)$ as inputs. The ILS problem and ILS algorithms are described briefly in Appendix C. This thesis uses a modified version of MILES [54] which returns both the optimal integer set, $\mathbf{N}_{opt}(k)$, and a tight lower bound on the probability that the integer set is correct, $P_{low}(k)$.

2. Once the optimal integer set is determined, the expected value of the real-valued portion of the state can be determined through the equation

$$E\left[\mathbf{x}(k)\right] = R_{xx}^{-1}(k)\left(\mathbf{z}_x(k) - R_{xN}(k)\mathbf{N}_{opt}(k)\right) \tag{4.3}$$

3. The quaternion elements of the state must be updated in a second step, since they are not represented directly in the state measurement equations. Their corresponding differential quaternions, which were computed in Eq. 4.3, are used to update the quaternions through Eq. 3.6. The differential quaternions must also be zeroed out in the state measurement equations so that this update is only performed once. This is accomplished for each differential quaternion through the equation

   $$\mathbf{z}'_x(k) = \mathbf{z}_x(k) - R_{x\delta e}(k)E\left[\delta\mathbf{e}\right] \tag{4.4}$$

   where $R_{x\delta e}(k)$ is the matrix containing the columns of $R_{xx}(k)$ corresponding to the differential quaternion. Updating the quaternions this way after every measurement update and propagation step prevents the differential quaternions from becoming large and violating the small angle assumption.

4. The covariance matrix can be computed through the equation

   $$P(k) = \left(\begin{bmatrix} R_{xx}(k) & R_{xN}(k) \\ 0 & R_{NN}(k) \end{bmatrix}^T \begin{bmatrix} R_{xx}(k) & R_{xN}(k) \\ 0 & R_{NN}(k) \end{bmatrix} \right)^{-1} \tag{4.5}$$

4.1.2 Initialization

The elements of the filter state are initialized as follows

$\mathbf{x}_{ECEF}^B$ and $\mathbf{v}_{ECEF}^B$ are initialized from the pseudorange-based navigation solution already computed by the mobile GPS receiver.

$\mathbf{b}_a$ is initialized to zero.

$\mathbf{q}^{B}_{ECEF}$ is initialized with the IMU's estimate of attitude.

$\mathbf{x}^{V}_{ECEF}$, $\mathbf{q}^{ECEF}_{V}$, and λ are initialized by comparing the visual SLAM solution to the coupled CDGPS and INS solution, which must be computed first, over the entire dataset. First, the quaternion $\mathbf{q}^{ECEF}_{V}$ can be computed as the difference between the attitude estimate from the visual SLAM solution and the coupled CDGPS and INS solution at a particular time. Second, the range to the reference GPS antenna can be plotted for both solutions based on initial guesses for $\mathbf{x}^{V}_{ECEF}$ and λ of $\mathbf{x}^{B}_{ECEF}$ and 1 and the value for $\mathbf{q}^{ECEF}_{V}$ that was already determined. After subtracting out the mean range from both solutions, the scale-factor λ can be computed as the ratio of amplitudes of the two traces. This assumes that the navigation system moved at some point during the dataset. Third, the position $\mathbf{x}^{V}_{ECEF}$ can be computed as the difference between the ECEF positions of the two solutions at a particular time.

$\mathbf{N}$ is initialized to zero.

4.2 Measurement Update Step

Measurements are grouped by subsystem and processed in the measurement update step in the order they arrive using the models developed in Chapter 3. Table 4.1 provides a list of the equations for the measurement models as a reference. The measurement update step proceeds in the same fashion as shown in Appendix B. A summary of this procedure is as follows

1. The linearized measurements are formed by subtracting the expected value of the measurements based on the a priori state and the non-linear measurement

Table 4.1: List of Equations for the Measurement Models

Subsystem	Measurement	Non-linear Model	Linearized Model		Covariance
		$h(\cdot)$	H_x	H_N	R
CDGPS	double-differenced pseudorange	Eq. 3.18	Eq. 3.28	0	Eq. 3.32
	double-differenced carrier-phase	Eq. 3.19	Eq. 3.28	Eq. 3.29	Eq. 3.33
INS	attitude estimate	Eq. 3.37	Eq. 3.40	0	Eq. 3.41
VNS	position estimate	Eq. 3.42	Eq. 3.45	0	Eq. 3.43
	attitude estimate	Eq. 3.46	Eq. 3.50	0	Eq. 3.51

model from the actual measurements. Equation numbers for the non-linear measurement models are listed in Table 4.1 for each measurement.

2. The linearized measurements and measurement models are then normalized using the Cholesky factorization of the inverse of the measurement covariance. Equation numbers for the linearized measurement models and measurement covariances are listed in Table 4.1 for each measurement.

3. The a priori estimate $\bar{\mathbf{x}}(k)$ is subtracted out of the state measurement equations to obtain the a priori delta-state measurement equations as

$$\begin{bmatrix} \delta\bar{\mathbf{z}}_x(k) \\ \bar{\mathbf{z}}_N(k) \end{bmatrix} = \begin{bmatrix} \bar{R}_{xx}(k) & \bar{R}_{xN}(k) \\ 0 & \bar{R}_{NN}(k) \end{bmatrix} \begin{bmatrix} \delta\mathbf{x}(k) \\ \mathbf{N} \end{bmatrix} + \begin{bmatrix} \mathbf{w}_x(k) \\ \mathbf{w}_N(k) \end{bmatrix} \tag{4.6}$$

   where $\delta\bar{\mathbf{z}}_x(k)$ is given by

$$\delta\bar{\mathbf{z}}_x(k) = \bar{\mathbf{z}}_x(k) - \bar{R}_{xx}(k)\bar{\mathbf{x}}(k) \tag{4.7}$$

4. The normalized measurement equations are stacked above Eq. 4.6. Using a QR factorization, the a posteriori delta-state measurement equations are then obtained in the same form as Eq. 4.6.

5. Adding back in the a priori estimate $\bar{\mathbf{x}}(k)$ to the a posteriori delta-state measurement equations results in the a posteriori state measurement equations in the same form as Eq. 4.2.

6. The a posteriori state and state covariance are then determined through the procedure specified in Sec. 4.1.1.

4.2.1 Handling Changes in the Tracked Satellites

Before performing a CDGPS measurement update, the satellites tracked by the reference and mobile GPS receivers are checked to see if the reference satellite should be changed or if any satellites should be dropped from or added to the list of satellites used in the measurement update. These changes necessitate modifications to the a priori state measurement equations prior to the CDGPS measurement update to account for changes in the definition of the integer ambiguity vector.

4.2.1.1 Changing the Reference Satellite

To obtain the lowest possible covariance for the double-differenced measurements, the reference satellite should be chosen as the satellite with the largest carrier-to-noise ratio. This roughly corresponds to the satellite at the highest elevation for most GPS antenna gain patterns. The highest elevation satellite will change as satellite geometry changes. Thus, a procedure for changing the reference satellite is desired. It is assumed that the new reference satellite was already in the list of tracked satellites before this measurement update step.

Before swapping the reference satellite, the portion of the a priori state measurement equations corresponding to the integer ambiguities is given as

$$\begin{aligned}\bar{\mathbf{z}}_N(k) &= \begin{bmatrix} \bar{z}^1_N(k) \\ \vdots \\ \bar{z}^i_N(k) \\ \vdots \\ \bar{z}^M_N(k) \end{bmatrix} = \bar{R}_{NN}(k)\mathbf{N} + \mathbf{w}_N(k) \\ &= \begin{bmatrix} \bar{R}^{11}_{NN}(k) & \cdots & \bar{R}^{1i}_{NN}(k) & \cdots & \bar{R}^{1M}_{NN}(k) \\ 0 & \ddots & \vdots & & \vdots \\ 0 & 0 & \bar{R}^{ii}_{NN}(k) & \cdots & \bar{R}^{iM}_{NN}(k) \\ 0 & 0 & 0 & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \bar{R}^{MM}_{NN}(k) \end{bmatrix} \begin{bmatrix} N^{10} \\ \vdots \\ N^{i0} \\ \vdots \\ N^{M0} \end{bmatrix} + \mathbf{w}_N(k)\end{aligned} \tag{4.8}$$

where the ith SV is the new reference satellite. Recall that the integer ambiguities can be decomposed into

$$N^{j0} = N^j - N^0, \text{ for } j = 1, \ldots, M \tag{4.9}$$

where N^j is the real-valued ambiguity on the single-differenced carrier-phase measurement for the jth SV. Therefore, the integer ambiguities with the ith SV as the reference can be related to the integer ambiguities with the original reference SV through the equation

$$N^{ji} = \begin{cases} N^{j0} - N^{i0} & ; j \neq 0, i \\ -N^{i0} & ; j = 0 \end{cases} \tag{4.10}$$

Using this relation, Eq. 4.8 can be rewritten with integer ambiguities referenced to the ith SV by modifying $\bar{R}_{NN}(k)$ and $\mathbf{N}$ as

$$
\begin{aligned}
\bar{\mathbf{z}}_N(k) &= \bar{R}'_{NN}(k)\mathbf{N}' + \mathbf{w}_N(k) \\
&= \begin{bmatrix}
\bar{R}^{11}_{NN}(k) & \cdots & \bar{R}^{1(i-1)}_{NN}(k) & \bar{R}^{10}_{NN}(k) & \bar{R}^{1(i+1)}_{NN}(k) & \cdots & \bar{R}^{1M}_{NN}(k) \\
0 & \ddots & \vdots & \vdots & \vdots & & \vdots \\
0 & 0 & \bar{R}^{(i-1)(i-1)}_{NN}(k) & \bar{R}^{(i-1)0}_{NN}(k) & \bar{R}^{(i-1)(i+1)}_{NN}(k) & \cdots & \bar{R}^{(i-1)M}_{NN}(k) \\
0 & 0 & 0 & \bar{R}^{00}_{NN}(k) & \bar{R}^{0(i+1)}_{NN}(k) & \cdots & \bar{R}^{0M}_{NN}(k) \\
0 & 0 & 0 & \bar{R}^{(i+1)0}_{NN}(k) & \bar{R}^{(i+1)(i+1)}_{NN}(k) & \cdots & \bar{R}^{(i+1)M}_{NN}(k) \\
0 & 0 & 0 & \vdots & 0 & \ddots & \vdots \\
0 & 0 & 0 & \bar{R}^{M0}_{NN}(k) & 0 & 0 & \bar{R}^{MM}_{NN}(k)
\end{bmatrix} \\
&\times \begin{bmatrix}
N^{1i} \\ \vdots \\ N^{(i-1)i} \\ N^{0i} \\ N^{(i+1)i} \\ \vdots \\ N^{Mi}
\end{bmatrix} + \mathbf{w}_N(k)
\end{aligned}
\tag{4.11}
$$

where all elements of $\bar{R}'_{NN}(k)$ are equal to the corresponding elements in $\bar{R}_{NN}(k)$ except for the ith column. Note that the terms in the ith row have been given different superscripts, but these terms are all equal to the corresponding elements of $\bar{R}_{NN}(k)$ except for $\bar{R}^{00}_{NN}(k)$. The elements of the ith column are given by the following equation

$$
\bar{R}^{j0}_{NN}(k) = \begin{cases} -\sum_{l=j}^{M} \bar{R}^{jl}_{NN}(k) & ; j \neq 0, i \\ -\sum_{l=i}^{M} \bar{R}^{il}_{NN}(k) & ; j = 0 \end{cases}
\tag{4.12}
$$

The cross-term between the real-valued and integer-valued portions of the state in the a priori state measurement equation, $\bar{R}_{xN}(k)$, must also be modified to account for this change in the integer ambiguity vector. Once again, only the ith column of $\bar{R}_{xN}(k)$ changes in value during this procedure. The elements of the ith column, using the same indexing scheme as before, are given by

$$\bar{R}_{xN}^{j0}(k) = -\sum_{l=1}^{M} \bar{R}_{xN}^{jl}(k) \tag{4.13}$$

4.2.1.2 Dropping Satellites

Whenever one of the GPS receivers is no longer tracking a particular satellite, the corresponding integer ambiguity must be removed from the filter state. If this satellite is the reference satellite, then the reference satellite must first be changed following the procedure in Sec. 4.2.1.1 so that only one integer ambiguity involves the measurements from the satellite to be removed. The satellite no longer tracked by both receivers will be referred to as the ith SV for the remainder of this section.

The integer ambiguity for the ith SV can be removed by first shifting the ith integer ambiguity to the beginning of the state and swapping columns in $\bar{R}_{xx}(k)$, $\bar{R}_{xN}(k)$, and $\bar{R}_{NN}(k)$ accordingly. After performing a QR factorization, the following equations are obtained

$$\begin{bmatrix} \bar{z}_N^{i\prime}(k) \\ \bar{\mathbf{z}}_x'(k) \\ \bar{\mathbf{z}}_N'(k) \end{bmatrix} = \begin{bmatrix} \bar{R}'_{N^{i0}N^{i0}}(k) & \bar{\mathbf{R}}'_{N^{i0}x}(k) & \bar{\mathbf{R}}'_{N^{i0}N}(k) \\ 0 & \bar{R}'_{xx}(k) & \bar{R}'_{xN}(k) \\ 0 & 0 & \bar{R}'_{NN}(k) \end{bmatrix} \begin{bmatrix} N^{i0} \\ \mathbf{x}(k) \\ \mathbf{N}' \end{bmatrix} + \begin{bmatrix} w_{N^{i0}}(k) \\ \mathbf{w}_x(k) \\ \mathbf{w}_N'(k) \end{bmatrix} \tag{4.14}$$

The first equation and the integer ambiguity N^{i0} can simply be removed with minimal effect on the rest of the state. If N^{i0} were real-valued, then there would be no information lost regarding the values of the other states by this method. Since N^{i0} is constrained to be an integer, some information is lost in this reduction. However, this method minimizes the loss in information to only that which is necessary for removal of the ambiguity from the state.

4.2.1.3 Adding Satellites

Adding a satellite is necessary whenever a new satellite is being tracked by both receivers. This procedure is much simpler than removing satellites from the state, since all that is necessary is to append the new ambiguity to the state and add a column of zeros and a row containing the prior to the a priori state measurement equations. Since no a priori information is available about the integer ambiguity for the new satellite, a defuse prior is used in its place in the a priori state measurement equations. The defuse prior assumes that the new integer ambiguity has an expected value of 0 and infinite variance, which can be represented with a 0 in information form. The resulting appended a priori state measurement equations are

$$\begin{aligned}\begin{bmatrix} \bar{\mathbf{z}}_x(k) \\ \bar{\mathbf{z}}_N(k) \\ 0 \end{bmatrix} &= \begin{bmatrix} \bar{R}_{xx}(k) & \bar{R}_{xN}(k) & 0 \\ 0 & \bar{R}_{NN}(k) & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x}(k) \\ \mathbf{N} \\ N^{(M+1)0} \end{bmatrix} + \begin{bmatrix} \mathbf{w}_x(k) \\ \mathbf{w}_N(k) \\ w_{N^{(M+1)0}}(k) \end{bmatrix} \\ = \begin{bmatrix} \bar{\mathbf{z}}_x(k) \\ \bar{\mathbf{z}}'_N(k) \end{bmatrix} &= \begin{bmatrix} \bar{R}_{xx}(k) & \bar{R}'_{xN}(k) \\ 0 & \bar{R}'_{NN}(k) \end{bmatrix} \begin{bmatrix} \mathbf{x}(k) \\ \mathbf{N}' \end{bmatrix} + \begin{bmatrix} \mathbf{w}_x(k) \\ \mathbf{w}'_N(k) \end{bmatrix}\end{aligned} \tag{4.15}$$

4.3 Propagation Step

Between measurement updates, the state measurement equations are propagated forward in time using either the INS or INS-free dynamics model derived in Chapter 3, depending on whether or not accelerometer measurements from the IMU are available. A propagation step is triggered by either an accelerometer measurement or a measurement update at a different time from the time index of the current filter state. Table 4.2 provides a list of equations for the dynamics models as a reference. The propagation step proceeds in the same fashion as shown in Appendix B. A summary of this procedure is as follows

Table 4.2: List of Equations for the Dynamics Models

Type	Difference Equation	State Transition Matrix	Process Noise Covariance
	$\mathbf{x}(k+1)$	$F(k)$	$Q(k)$
INS	Eq. 3.57	Eq. 3.59	Eq. 3.60
INS-Free	Eq. 3.71	Eq. 3.72	Eq. 3.73

1. The a priori estimate $\bar{\mathbf{x}}(k+1)$ is computed from the state difference equation evaluated at the a posteriori estimate $\hat{\mathbf{x}}(k)$ and the time interval of the propagation step, Δt. Equation numbers for the state difference equations are listed in Table 4.2 for both dynamics models.

2. The a posteriori state measurement equations at the beginning of the propagation interval are stacked below the process noise measurement equation given as

$$\mathbf{z}_v(k) = 0 = R_{vv}(k)\mathbf{v}(k) + \mathbf{w}_v(k) \tag{4.16}$$

   where $R_{vv}(k)$ is the Cholesky factorization of the inverse of the process noise covariance. Equation numbers for the process noise covariances are listed in Table 4.2 for both dynamics models.

3. $\mathbf{x}(k+1)$ is substituted for $\mathbf{x}(k)$ in the stacked process noise and state measurement equations through the linearized dynamics equation. The linearized dynamics equation is simply the difference equation evaluated at the a posteriori estimate $\hat{\mathbf{x}}(k)$ plus the term $F(k)(\mathbf{x}(k) - \hat{\mathbf{x}}(k))$. Equation numbers for the state transition matrix, $F(k)$, are listed in Table 4.2 for both dynamics models.

4. Using a QR factorization, the a priori state measurement equations at the end of the propagation interval are obtained in the same form as Eq. 4.2. If the a

priori state covariance is desired, then it can be computed through the procedure specified in Sec. 4.1.1.

Chapter 5

Prototype Augmented Reality System

A prototype AR system based on the navigation filter defined in Chapters 3 and 4 was designed and built to demonstrate the accuracy of such a system. Figure 5.1 shows a picture of the prototype AR system, which is composed of a tablet computer attached to a sensor package. A webcam points out the side of the sensor package opposite from the tablet computer to provide a view of the real world that is displayed on the tablet computer and augmented with virtual elements. The tablet computer could thus be thought of as a "window" into the AR environment; a user looking "through" the tablet computer would see an augmented representation of the real world on the other side of the AR system. However, the navigation filter and augmented visuals are currently only implemented in post-processing. Therefore, the tablet computer simply acts as a data recorder at present. This prototype AR system is an advanced version of that presented in [47].

This chapter describes the hardware and software used for the sensor package in the prototype AR system. This sensor package can be divided into three navigation "subsystems", CDGPS, INS, and VNS, which are detailed separately in the following sections. For reference, a picture of the sensor package with each of the hardware components labeled is shown in Fig. 5.2. Each of the labeled components, except the Lithium battery, are detailed in the hardware section for their corresponding subsystem.

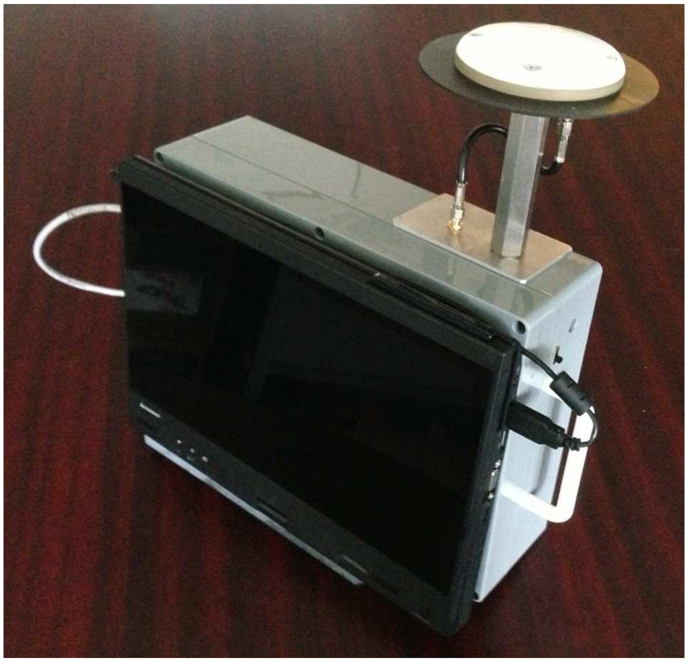

Figure 5.1: The assembled prototype augmented reality system.

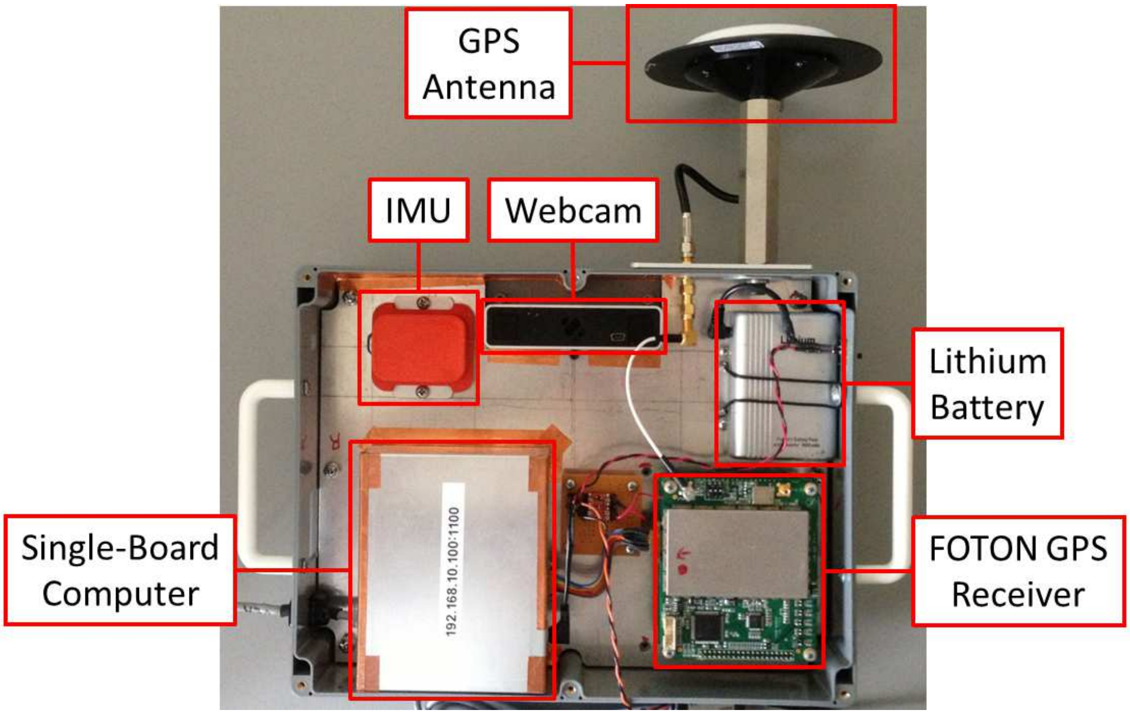

Figure 5.2: A labeled picture of the sensor package for the prototype augmented reality system.

5.1 CDGPS

The CDGPS subsystem is represented in the block diagram in Fig. 3.1 by the boxes encircled by a blue dashed line. The sensors for the CDGPS subsystem are the mobile GPS receiver and the reference GPS receiver, which is not part of the sensor package. The reference GPS receiver used for the tests detailed in Chapter 6 was a CASES software-defined GPS receiver developed by The University of Texas at Austin and Cornell University. CASES can report GPS observables and pseudorange-based navigation solutions at a configurable rate, which was set to 5 Hz for the prototype AR system. These data can be obtained from CASES over the internet. Further information on CASES can be found in [55]. For the tests detailed in Chapter 6, CASES operated on data collected from a high-quality Trimble antenna located at a surveyed location on the roof of the W. R. Woolrich Laboratories at The University of Texas at Austin. The mobile GPS receiver, which is part of the sensor package, is composed of the hardware and software described below.

5.1.1 Hardware

5.1.1.1 FOTON GPS Receiver

The mobile GPS receiver used for the prototype AR system was the FOTON software-defined GPS receiver developed by The University of Texas at Austin and Cornell University. FOTON is a dual-frequency receiver that is capable of tracking GPS L1 C/A and L2C signals, but only the L1 C/A signals were used in the prototype AR system. FOTON can be seen in the lower right-hand corner of Fig. 5.2. The workhorse of FOTON is a digital signal processor (DSP) running the GRID software receiver, which is described in Sec. 5.1.2.1.

5.1.1.2 Single-Board Computer

The single-board computer (SBC) is used for communications between FOTON and the tablet computer. FOTON sends data packets to the SBC over a serial interface. These data packets are then buffered by the SBC and sent to the tablet computer via Ethernet. The SBC is not strictly necessary and could be removed from the system in the future if a direct interface between FOTON and the tablet computer were created.

The SBC is located under the metal cover in the lower left-hand corner of Fig. 5.2. This metal cover was placed over the SBC because the SBC was emitting noise in the GPS band that was reaching the antenna and causing significant degradation of the received carrier-to-noise ratio. The addition of the metal cover largely eliminated this problem.

5.1.1.3 Antcom GPS Antenna

The GPS antenna used for the prototype AR system was a 3.5” GPS L1/L2 antenna from Antcom [56]. This antenna can be seen in the upper right-hand corner of Fig. 5.2. This antenna has good phase-center stability, which is necessary for CDGPS, but is admittedly quite large. Reducing the size of the antenna much below this while maintaining good phase-center stability is a difficult antenna design problem that has yet to be solved. Therefore, the size of the antenna is currently the largest obstacle to miniaturizing the sensor package for an AR system employing CDGPS.

5.1.2 Software

5.1.2.1 GRID Software Receiver

As mentioned in Sec. 5.1.1.1, the GRID software receiver, which was developed by The University of Texas at Austin and Cornell University, runs on the FOTON's DSP [57, 58]. GRID is responsible for

1. performing complex correlations between the digitized samples from FOTON's radio-frequency front-end at an intermediate frequency and local replicas of the GPS signals.
2. acquiring and tracking the GPS signals based on these complex correlations.
3. computing the GPS observables measurements and pseudorange-based navigation solutions.

GPS observables measurements and pseudorange-based navigation solutions can be output from GRID at a configurable rate, which was set to 5 Hz for the prototype AR system.

Carrier-phase cycle slips are a major problem in CDGPS-based navigation because cycle slips result in changes to the integer ambiguities on the double-differenced carrier-phase measurements. Thus, cycle slip prevention is paramount for CDGPS. GRID was originally developed for Ionospheric monitoring. As such, GRID has a scintillation robust PLL and databit prediction capability, which both help to prevent cycle slips [55].

5.2 Inertial Navigation System

The INS subsystem is represented in the block diagram in Fig. 3.1 by the boxes encircled by a red dashed line. The sensors for the INS subsystem are contained within a single IMU located on the sensor package. This IMU is detailed below.

5.2.1 Hardware

5.2.1.1 XSens MTi IMU

The IMU used for the prototype AR system was the XSens MTi, which can be seen in the center of the left-hand side of Fig. 5.2. This IMU is a complete gyro-enhanced attitude and heading reference system (AHRS). It houses four sensors, (1) a magnetometer, (2) a three-axis gyro, (3) a three-axis accelerometer, and (4) a thermometer. The MTi also has a DSP running a Kalman filter, referred to as the XSens XKF, that determines the attitude of the MTi relative to the north-west-up (NWU) coordinate system, which is converted to ENU for use in the navigation filter. The XKF is described in Sec. 5.2.2.1.

In addition to providing attitude, the MTi also provides access to the highly-stable, temperature-calibrated (via the thermometer and high-fidelity temperature models) magnetometer, gyro, and accelerometer measurements. The MTi can output these measurements and the attitude estimate from the XKF at a configurable rate, which was set to 100 Hz for the prototype AR system. In order to obtain a time stamp for the MTi data, the MTi measurements were triggered by FOTON, which also reported the GPS time the triggering pulse was sent.

5.2.2 Software

5.2.2.1 XSens XKF

As mentioned in Sec. 5.2.1.1, the XSens XKF is a Kalman filter that runs on the MTi's DSP and produces estimates of the attitude of the MTi relative to NWU. This Kalman filter determines attitude by ingesting temperature-calibrated (via the MTi's thermometer and high-fidelity temperature models) magnetometer, gyro, and accelerometer measurements from the MTi to determine magnetic North and the gravity vector. If the XKF is given magnetic declination, which can be computed from models of the Earth's magnetic field and the position of the system, then true North can be determined from magnetic North. Knowledge of true North and the gravity vector is sufficient for full attitude determination relative to NWU. This estimate of orientation is reported in the MTi specifications as accurate to better than 2^o RMS for dynamic operation. However, magnetic disturbances and long-term sustained accelerations can cause the estimates of magnetic North and the gravity vector respectively to develop biases.

5.3 Visual Navigation System

The VNS subsystem is represented in the block diagram in Fig. 3.1 by the boxes encircled by a green dashed line. The VNS subsystem uses video from a webcam located on the sensor package to extract navigation information via a stand-alone BA-based visual SLAM algorithm. This webcam and the visual SLAM software are detailed below.

5.3.1 Hardware

5.3.1.1 FV Touchcam N1 Webcam

The webcam used for the prototype AR system was the FV Touchcam N1, which can be seen in the center of Fig. 5.2. The Touchcam N1 is an HD webcam capable of outputting video in several formats and frame-rates including 720P-format video at 22 fps and WVGA-format video at 30 fps. The Touchcam N1 also has a wide angle lens with a 78.1^o horizontal field of view.

5.3.2 Software

5.3.2.1 Parallel Tracking and Mapping

The visual SLAM algorithm used for the prototype AR system was PTAM developed by Klein and Murray [45]. PTAM is capable of tracking thousands of point features and estimating relative pose up to an arbitrary scale-factor at 30 Hz frame-rates on a dual-core computer. Further details on PTAM can be found in Sec. 2.2.1.2 and [45].

Time alignment of the relative pose estimates from PTAM with GPS time was performed manually, since the webcam video does not contain time stamps traceable GPS time. This time alignment was performed by comparing the relative pose from PTAM and the coupled CDGPS and INS solution over the entire dataset. The initial guess for the GPS time of the first relative pose estimate from PTAM is taken as the GPS time of the first observables measurement of the dataset. The time rate offset is assumed to be zero, which is a reasonable assumption for short datasets. From a plot of the range to the reference GPS antenna assuming initial guesses for $\mathbf{x}_{ECEF}^{V}$, $\mathbf{q}_{V}^{ECEF}$, and λ of $\mathbf{x}_{ECEF}^{B}$, $\begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}^T$ and 1 respectively, the time offset between GPS time and the initial guess for PTAM's solution can be determined by aligning the changes in the range to the reference GPS receiver in time. Note that the traces in this plot will not align because $\mathbf{x}_{ECEF}^{V}$, $\mathbf{q}_{V}^{ECEF}$, and λ have yet to be determined. However, the times when the range to the reference GPS receiver changes can be aligned. Better guesses for $\mathbf{x}_{ECEF}^{V}$, $\mathbf{q}_{V}^{ECEF}$, and λ can be determined from the initialization procedure described in Sec. 4.1.2 once the data has been time aligned.

Chapter 6

Test Results for the Prototype Augmented Reality System

The prototype AR system presented in Chapter 5 was tested in several different modes of operation to demonstrate the accuracy and precision of the prototype AR system. These modes were CDGPS, coupled CDGPS and INS, and coupled CDGPS, INS, and VNS. Testing these modes incrementally allows for demonstration of the benefits of adding each additional navigation subsystem to the prototype AR system.

This chapter presents test results from the prototype AR system for each of these modes of operation and details the performance of the prototype AR system. The chapter begins by presenting results from the CDGPS subsystem operating alone in both static and dynamic scenarios. These results demonstrate the positioning accuracy and precision of the CDGPS subsystem. Next, results from the coupled CDGPS and INS mode are presented for the dynamic scenario. The addition of the INS provides both absolute attitude information and inertial measurements to smooth out the position solution between CDGPS measurements. The coupled CDGPS and INS solution is also compared to the VNS solution after determining the similarity transform between the V-frame and ECEF. Finally, results from the complete navigation system, which couples CDGPS, INS, and VNS, are given for the dynamic scenario. These results demonstrate significant improvement of performance over the coupled CDGPS and INS solution in both absolute positioning and absolute attitude.

6.1 Results for CDGPS Mode

In CDGPS mode, the prototype AR system only processes measurements from the CDGPS subsystem. Therefore, attitude cannot be estimated by the navigation filter in this mode. However, this mode is useful for demonstrating the positioning accuracy and precision attained by the CDGPS subsystem. The following sections present test results for both static and dynamic tests of the system in this mode.

6.1.1 Static Test Results

The static test was performed using two antennas, identified in Fig. 6.1, separated by a short baseline distance and located on top of the W. R. Woolrich Laboratories (WRW) at The University of Texas at Austin. This baseline distance between the two receivers was measured by tape measure to be approximately 21.155 m [47]. Twenty minutes of GPS observables data was collected at 5 Hz from receivers connected to each of the antennas. This particular dataset had data from 11 GPS satellites with one of the satellites rising 185.2 s into the dataset and another setting 953 s into the dataset.

Figure 6.2 shows a lower bound on the probability that the integer ambiguities have converged to the correct solution for the first 20 s of the static test. Details on how to compute this lower bound are provided in Appendix C. A probability of 0.999 was used as the metric for declaring that the integer ambiguities have converged to the correct values and was attained 15.8 s into the test. The integer ambiguities actually converged to the correct values and remained at the correct values after the first 10.6 s of the test, even with a satellite rising and another setting during the dataset. This demonstrates that the methods for handling adding and dropping of integer ambiguities to/from the filter state outlined in Sec. 4.2.1 are performing as

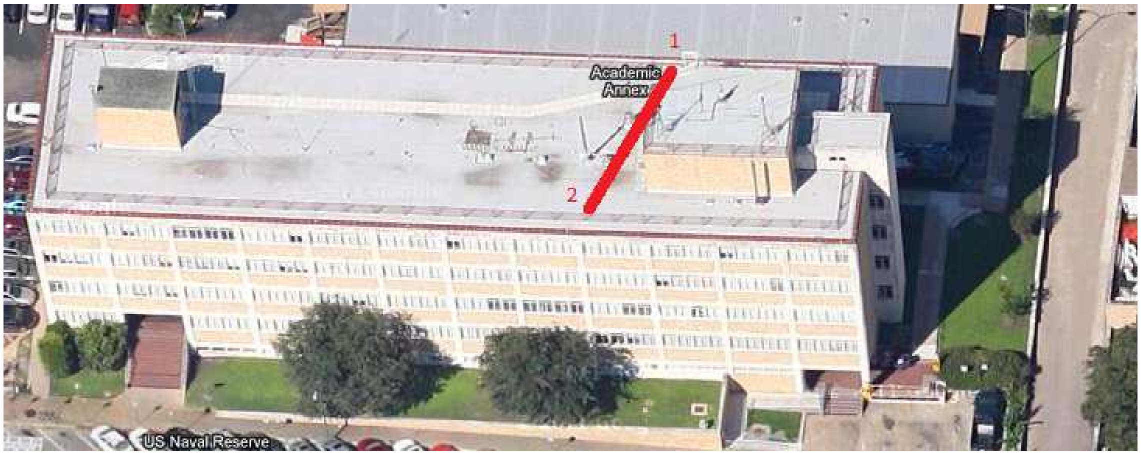

Figure 6.1: Picture showing the approximate locations of the two antennas used for the static test. Antenna 1 is the reference antenna, which is also used as the reference antenna for the dynamic test.

expected.

Although the true convergence of the integer ambiguities occurred prior to reaching the 0.999 lower bound probability metric for this test, other tests not presented in this thesis revealed that this is all too often not the case for the CDGPS algorithm as implemented for this thesis. This is likely due to ignoring the time-correlated multipath errors on the double-differenced GPS observables measurements. The GPS antennas and receivers used for the prototype system do not have good multipath rejection capabilities. Therefore, future versions of the prototype system will need to better handle multipath errors on the double-differenced GPS observables measurements to enable confidence in the convergence metric. This could be accomplished through the use of receiver processing techniques to mitigate the effects of multipath on the GPS observables.

A trace of the East and North coordinates of the mobile antenna relative to the reference antenna, whose location is known in ECEF, as estimated by the prototype AR system in CDGPS mode is shown in Fig. 6.3 for the static test. Only position esti-

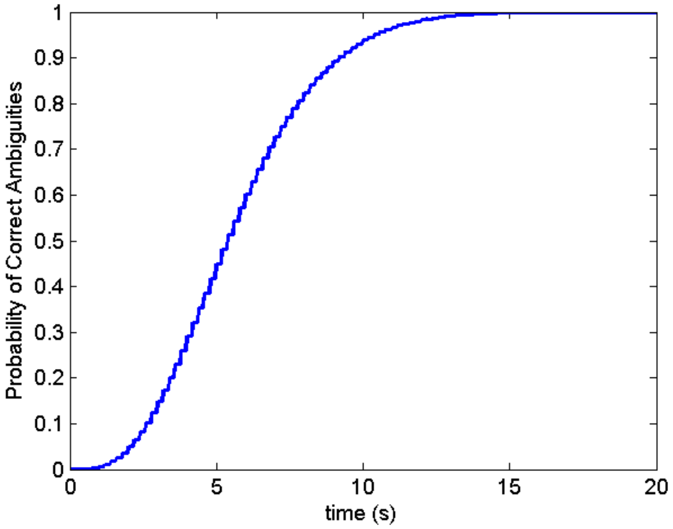

Figure 6.2: This plot shows a lower bound on the probability that the integer ambiguities are correct as a function of time for the static test. Only the first 20 s of the test are shown, since the probability is 1 for the remainder of the test.

mates from after the integer ambiguities were declared converged are shown in Fig. 6.3. These position estimates all fit inside a 2 cm by 2 cm by 4 cm rectangular prism in ENU centered on the mean position, which demonstrates the precision of the CDGPS subsystem. The mean of the position estimates expressed in the ENU-frame centered on the reference antenna was $E\left[\mathbf{x}_{ENU}^{B}\right] = \left[\begin{array}{ccc} -16.8942 & -11.3368 & -5.8082 \end{array}\right]$ m. This results in an estimated baseline distance of 21.1583 m, which is almost exactly the measured baseline distance of 21.155 m. This difference between the estimated and measured baselines is well within the expected error of the measured baseline, thus demonstrating the accuracy of the CDGPS subsystem.

To further illustrate the precision of the CDGPS subsystem, Fig. 6.4 shows plots of the deviations (in blue) of the East, North, and Up position estimates from the mean over the entire dataset from after the integer ambiguities were declared converged. The +/- 1 standard deviation bounds are also shown in Fig. 6.4 based

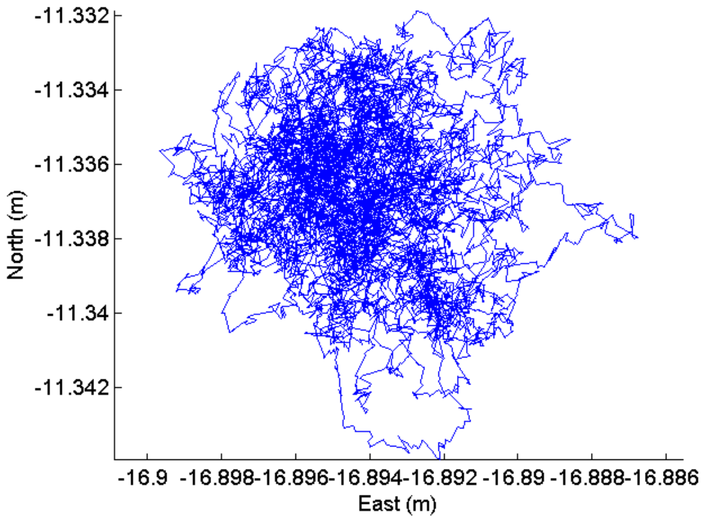

Figure 6.3: This plot shows a trace of the East and North position of the mobile antenna as estimated by the prototype AR system in CDGPS mode for the static test from after the integer ambiguities were declared converged. The position estimates are expressed in the ENU frame centered on the reference antenna.

on both the filter covariance estimate (in red) and the actual standard deviation (in green) of the position estimates over the entire dataset. The actual standard deviations were $\sigma_E = 0.002$ m, $\sigma_N = 0.002$ m, and $\sigma_U = 0.0051$ m. As can be seen from Fig. 6.4, the filter covariance estimates closely correspond to the actual covariance of the data over the entire dataset, which is a highly desirable quality that arises because the noise on the GPS observables measurements is well modeled.

6.1.2 Dynamic Test Results

The dynamic test was performed using the same reference antenna, identified as 1 in Fig. 6.1, as the static test. The prototype AR system, which was also on the roof of the WRW for the entire dataset, remained stationary for the first four and a half minutes of the dataset to ensure that the integer ambiguities could converge before the system began moving. This is not strictly necessary, but ensured that good

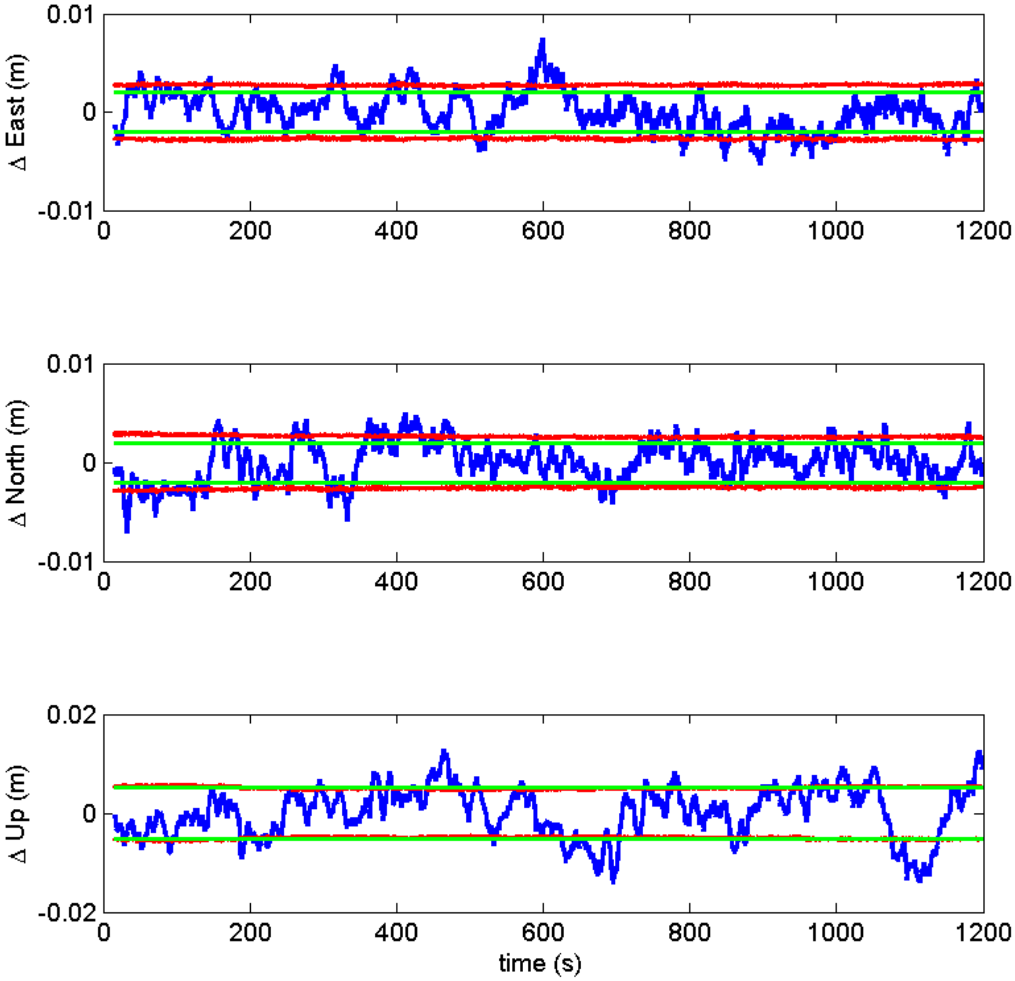

Figure 6.4: These three plots show the East (top), North (middle), and Up (bottom) deviations about the mean of the position estimate from the prototype AR system in CDGPS mode for the static test. The deviations, shown in blue, are expressed in the ENU frame centered on the reference antenna. The green and red lines in each plot represent +/- 1 standard deviation from the mean based on the estimated positions over the entire test and the navigation filter covariance as a function of time respectively.

data was collected for analysis. After this initial stationary period, the prototype AR system was walked around the front of a wall for one and a half minutes before returning to its original location. Virtual graffiti was to be augmented onto the real-world view of the wall provided by the prototype AR system's webcam. This approximately 6 minute dataset contained data from 10 GPS satellites with one of the satellites rising 320.4 s into the dataset.

One of the satellites was excluded from the dataset because it was blocked by the wall when the system began moving, which caused a number of cycle slips prior to the mobile GPS receiver loosing lock on the satellite. Most cycle slips are prevented by masking out low elevation satellites, but environmental blockage such as this can pose significant problems. Therefore, a cycle slip detection and recovery algorithm would be useful for the AR system and is an area of future work.

Figure 6.5 shows a lower bound on the probability that the integer ambiguities have converged to the correct solution for the first 40 s of the dynamic test. The integer ambiguities were declared converged by the filter after a probability of 0.999 was attained 31.4 s into the test. This took almost twice as long as for the static test because this dataset only had data from 8 GPS satellites during this interval while the static test had data from 10 GPS satellites. The integer ambiguities actually converged to the correct value and remained at the correct value after the first 10.6 s of the test, which only coincidentally corresponds to actual convergence time for the static test.

A trace of the East and North coordinates of the mobile antenna relative to the reference antenna as estimated by the prototype AR system in CDGPS mode is shown in Fig. 6.6 for the dynamic test. Only position estimates from after the integer ambiguities were declared converged are shown in Fig. 6.6. The system began

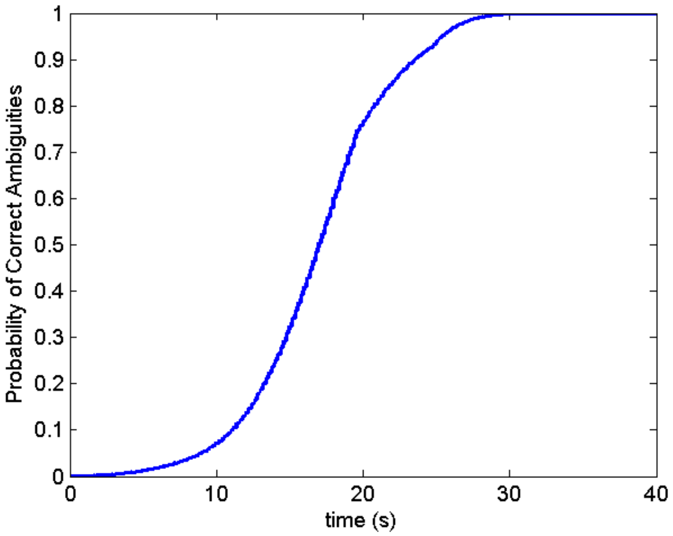

Figure 6.5: This plot shows a lower bound on the probability that the integer ambiguities are correct as a function of time for the dynamic test. Only the first 40 s of the test are shown, since the probability is 1 for the remainder of the test.

at a position of roughly [-43.077, -5.515, -6.08] m before being picked up, shaken from side to side a few times, and carried around while looking toward a wall that was roughly north of the original location. Position estimates were output from the navigation filter at 30 Hz, while GPS measurements were only obtained at 5 Hz. The INS-free dynamics model described in Sec. 3.3.2 is used to propagate the solution between GPS measurements. This dynamics model knows nothing about the actual dynamics of the system. Therefore, the positioning accuracy suffers during motion of the system. The position estimate is also not very smooth, which may cause any augmented visuals based on this position estimate to shake relative to the real world. Therefore, a better dynamics model is desired in order to preserve the illusion of realism of the augmented visuals during motion.

To illustrate the precision of the estimates, Figs. 6.7 and 6.8 show the standard deviations of the ENU position estimates of the mobile antenna based on the filter

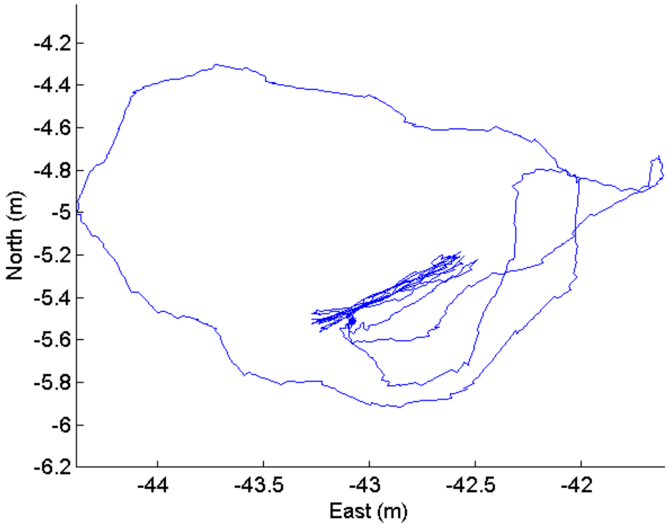

Figure 6.6: This plot shows a trace of the East and North position of the mobile antenna as estimated by the prototype AR system in CDGPS mode for the dynamic test from after the integer ambiguities were declared converged. The position estimates are expressed in the ENU frame centered on the reference antenna.

covariance estimates from the prototype AR system in CDGPS mode from just before and just after CDGPS measurement updates respectively. Taking standard deviations of the position estimates from these two points in the processing demonstrates the best and worst case standard deviations for the system. These standard deviations are an order of magnitude larger than those for the static test because the standard deviation of the velocity random walk term in the dynamics model was increased from 0.001 m/s$^{3/2}$ (roughly stationary) to 0.5 m/s$^{3/2}$, which is a reasonable value for human motion. Velocity random walk essentially models the acceleration as zero-mean Gaussian white noise with an associated covariance. This is typically a good model for human motion provided that the associated covariance is representative of the true motion. Assuming that the chosen velocity random walk covariance is representative of the true motion, the position estimates are accurate to centimeter-level at all times, as can be seen in Figs. 6.7 and 6.8.

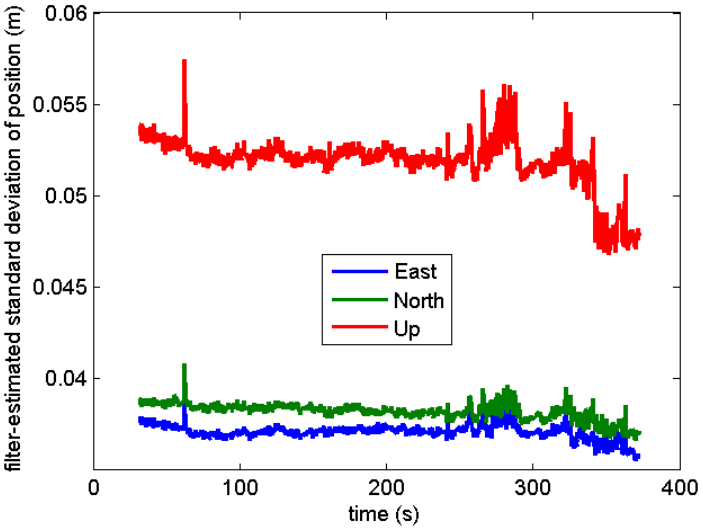

Figure 6.7: This plot shows the standard deviations of the East (blue), North (green), and Up (red) position estimates of the mobile antenna based on the filter covariance estimates from the prototype AR system in CDGPS mode for the dynamic test from just before CDGPS measurement updates. The standard deviations are expressed in the ENU frame centered on the reference antenna.

6.2 Results for Coupled CDGPS and INS Mode

The addition of an INS to the system allows for determination of attitude relative to ECEF and a better dynamics model that leverages accelerometer measurements to propagate the state between CDGPS measurements. This mode produces precise and globally-referenced pose estimates that can be used for AR. However, the IMU attitude solution is susceptible local magnetic disturbances and long-term sustained accelerations, which may cause significant degradation of performance. This will be illustrated in the following sections, which provide results for the dynamic test described in Sec. 6.1.2.

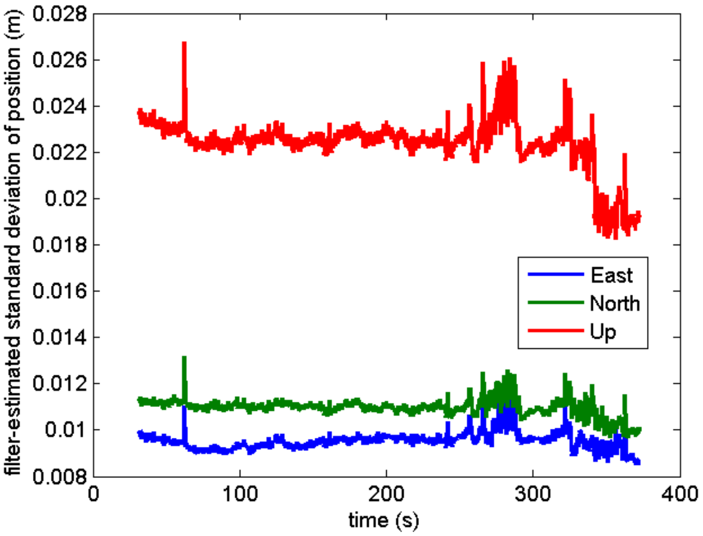

Figure 6.8: This plot shows the standard deviations of the East (blue), North (green), and Up (red) position estimates of the mobile antenna based on the filter covariance estimates from the prototype AR system in CDGPS mode for the dynamic test from just after CDGPS measurement updates. The standard deviations are expressed in the ENU frame centered on the reference antenna.

6.2.1 Positioning Results

A trace of the East and North coordinates of the mobile antenna relative to the reference antenna as estimated by the prototype AR system in coupled CDGPS and INS mode is shown in Fig. 6.9 for the dynamic test. Only position estimates from after the integer ambiguities were declared converged, which occurred at the same time as in CDGPS mode, are shown in Fig. 6.9. From comparing Figs. 6.9 and 6.6, it can be seen that the addition of the INS resulted in a much more smoothly varying estimate of the position. While accuracy of the position estimates is very important for AR to reduce the registration errors, accurate position estimates that have a jerky trajectory will result in virtual elements that shake relative to the background. If the magnitude of this shaking is too large, then the illusion of realism of the virtual object will be broken.

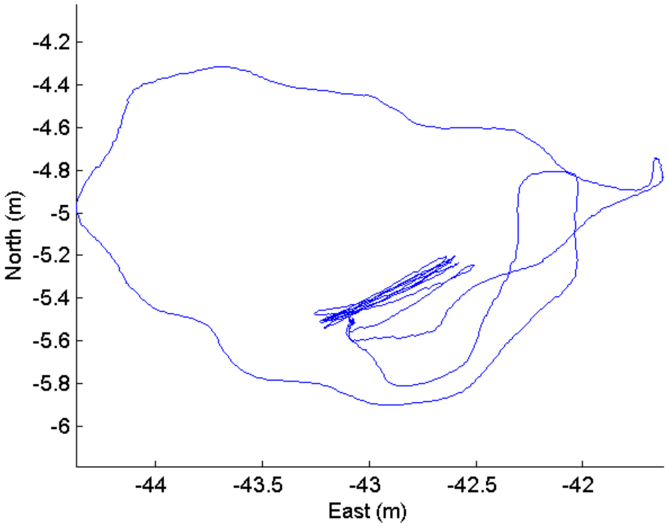

Figure 6.9: This plot shows a trace of the East and North position of the mobile antenna as estimated by the prototype AR system in coupled CDGPS and INS mode for the dynamic test from after the integer ambiguities were declared converged. The position estimates are expressed in the ENU frame centered on the reference antenna.

To illustrate the precision of the position estimates, Figs. 6.10 and 6.11 show the standard deviations of the ENU position estimates of the IMU based on the filter covariance estimates from the prototype AR system in coupled CDGPS and INS mode from just before and just after CDGPS measurement updates respectively. The standard deviations taken from before the CDGPS measurement updates for this mode are significantly smaller than those from the CDGPS mode, shown in Fig. 6.7, as expected. This is due to the improvement in the dynamics model of the filter enabled by the accelerometer measurements from the IMU. In fact, the reduction in process noise enabled by the IMU accelerometer measurements lowers the standard deviations to the point that the standard deviations taken from before the CDGPS measurement updates for this mode are slightly smaller than those from after the CDGPS measurement updates for CDGPS mode, shown in Fig. 6.8.

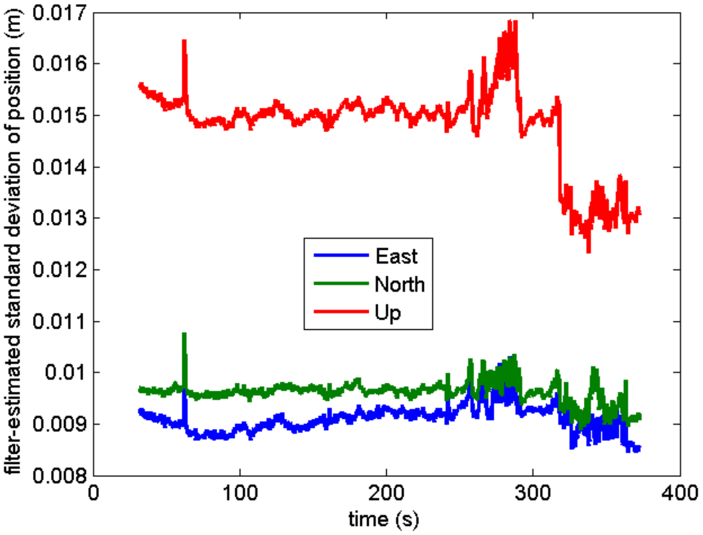

Figure 6.10: This plot shows the standard deviations of the East (blue), North (green), and Up (red) position estimates of the IMU based on the filter covariance estimates from the prototype AR system in coupled CDGPS and INS mode for the dynamic test from just before CDGPS measurement updates. The standard deviations are expressed in the ENU frame centered on the reference antenna.

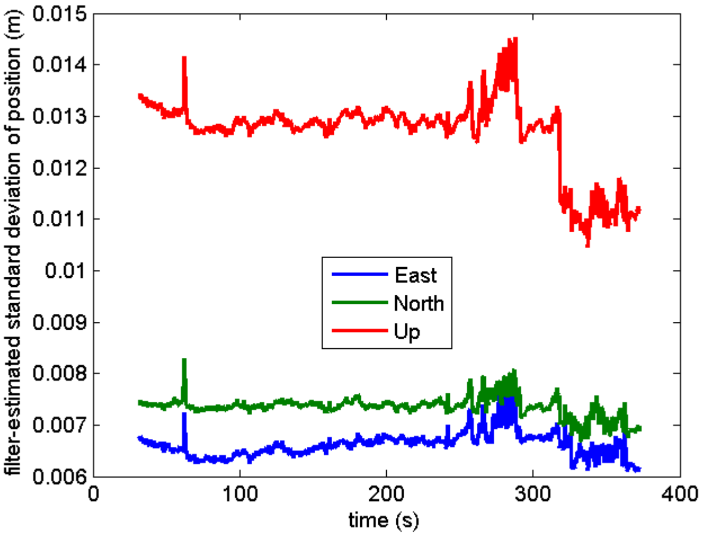

Figure 6.11: This plot shows the standard deviations of the East (blue), North (green), and Up (red) position estimates of the IMU based on the filter covariance estimates from the prototype AR system in coupled CDGPS and INS mode for the dynamic test from just after CDGPS measurement updates. The standard deviations are expressed in the ENU frame centered on the reference antenna.

6.2.2 Attitude Results

The attitude estimates, expressed as standard yaw-pitch-roll Euler angle sequences, from the prototype AR system in coupled CDGPS and INS mode are shown in Fig. 6.12 for the dynamic test. It was discovered during analysis of this dataset that the IMU estimated attitude had a roughly constant 26.5° bias in yaw, likely due to a magnetic disturbance throwing off the IMU's estimate of magnetic North. The discovery of the bias is detailed in Sec. 6.3.1. This bias was removed from the IMU data and the dataset reprocessed such that all results presented in this thesis do not contain this roughly constant portion of the bias. In future versions of the prototype AR system, it is thus desirable to eliminate the need of a magnetometer to estimate attitude. This can be accomplished through a tighter coupling of CDGPS and VNS, as explained in Chapter 2.

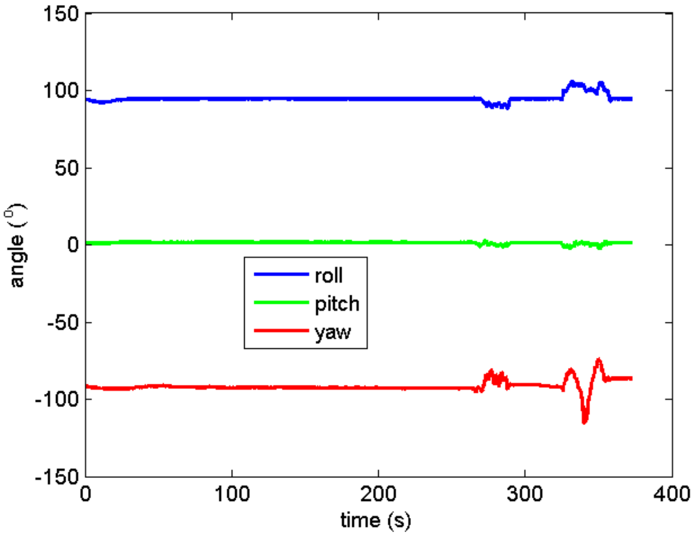

Figure 6.12: This plot shows the attitude estimates from the prototype AR system in coupled CDGPS and INS mode for the dynamic test. Attitude is expressed here in the standard yaw-pitch-roll Euler angles for ease of interpretation.

To illustrate the precision of the attitude estimates, Fig. 6.13 shows the expected standard deviation of the rotation angle between the true attitude and the estimated attitude from the prototype AR system in coupled CDGPS and INS mode for the dynamic test. This is computed from the filter covariance estimate based on the definition of the quaternion, given in Appendix A, as follows

$$\theta(k) = 2 \arcsin\left(\sqrt{P_{(\delta e_1,\delta e_1)}(k) + P_{(\delta e_2,\delta e_2)}(k) + P_{(\delta e_3,\delta e_3)}(k)}\right) \tag{6.1}$$

where $P_{(\delta e_1,\delta e_1)}(k)$, $P_{(\delta e_2,\delta e_2)}(k)$, and $P_{(\delta e_3,\delta e_3)}(k)$ are the diagonal elements of the filter covariance estimate corresponding to the elements of the differential quaternion. This shows that the filter believes the error in its estimate of attitude has a standard deviation of no worse than 1.35° at any time. It should be noted that since no truth data is available it is not possible to verify the accuracy of the attitude estimate, but consistency, or lack of consistency, between this solution and the VNS solution is

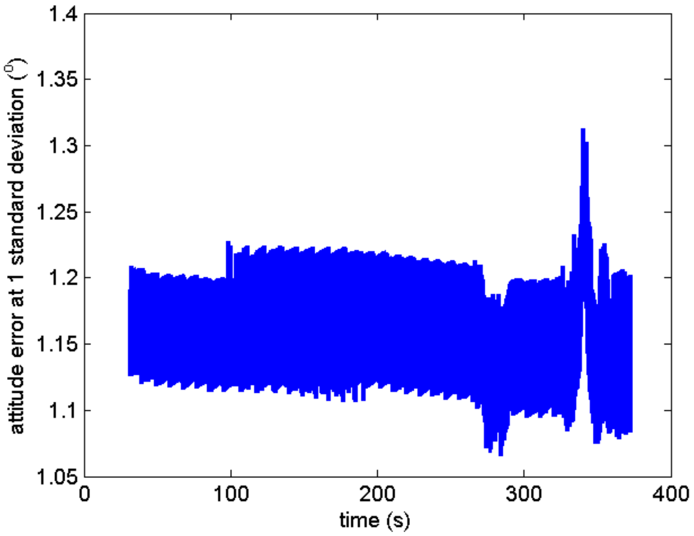

Figure 6.13: This plot shows the expected standard deviation of the rotation angle between the true attitude and the estimated attitude based on the filter covariance estimates from the prototype AR system in coupled CDGPS and INS mode for the dynamic test.

shown in Sec. 6.3.1.

6.3 Results for Coupled CDGPS, INS, and VNS Mode

The addition of a VNS to the system provides a second set of measurements of both position and attitude. The additional attitude measurement is of particular consequence because VNS attitude measurements are not susceptible to magnetic disturbances like the INS attitude measurements. The loose coupling of the VNS to both CDGPS and INS implemented in this prototype AR system does enable improvement of the estimates of both absolute position and absolute attitude over the coupled CDGPS and INS solution. However, this requires that the prototype AR system estimate the similarity transform between ECEF and the V-frame. In the future, this intermediate V-frame could be eliminated through a tighter coupling of the VNS and CDGPS, as explained in Chapter 2.

This section begins by demonstrating that the VNS solution is consistent, except for a roughly constant bias in the IMU attitude estimate, with the coupled CDGPS and INS solution for the dynamic test. Then, the results for the prototype AR system in coupled CDGPS, INS, and VNS mode are provided for the dynamic test described in Sec. 6.1.2.

6.3.1 Consistency of Solutions

Before coupling the VNS to the CDGPS and INS solution, consistency between the two solutions can be demonstrated with a single constant estimate of the similarity transform between ECEF and the V-frame. While this does not prove the accuracy of either solution in an absolute sense, consistency of the two solutions demonstrates accurate reconstruction of the trajectory of the prototype AR system. Combining this with the proven positioning accuracy of the CDGPS-based position estimates and motion of the system results in verification of the accuracy of the complete pose estimates. To be more specific, a bias in the attitude estimates from the IMU would find its way into the estimate of the similarity transform between ECEF and the V-frame and, for the procedure for determining this similarity transform described in Sec. 4.1.2, would result in a rotation of the VNS position solution about the initial location of the prototype AR system. This is how the bias in the IMU's estimate of yaw was discovered.

The estimate of the similarity transform between ECEF and the V-frame is determined through the initialization procedure described in Sec. 4.1.2. This procedure may not result in the best estimate of the similarity transform, but it will be close to the best estimate. The VNS solution after transformation to absolute coordinates through the estimate of the similarity transform will be referred to as the calibrated VNS solution for the remainder of this section.

Figure 6.14 shows the norm of the difference between the position of the webcam as estimated by the prototype AR system in coupled CDGPS and INS mode and the calibrated VNS solution from PTAM for the dynamic test. During stationary portions of the dataset, the position estimates agree to within 2 cm of one another at all times after an initial settling period. During periods of motion, the position estimates still agree to within 5 cm for more than 90% of the time. This larger difference between position estimates during motion occurs primarily because errors in the estimate of the similarity transform between ECEF and the V-frame are more pronounced during motion. Even with these errors, centimeter-level agreement of the position estimates between the two solutions is obtained at all times. The agreement might be even better if a more accurate estimate of the similarity transform between ECEF and the V-frame were determined.

Figure 6.15 shows the rotation angle between the attitude of the webcam as estimated by the prototype AR system in coupled CDGPS and INS mode and the calibrated VNS solution from PTAM for the dynamic test. The attitude estimates agree to within a degree for the entirety of the stationary period of the dataset. Once the system begins moving, the attitude estimates diverge from one another. By the end of the dataset, the two solutions only agree to within about 3°. This divergence was a result of the IMU trying to correct the 26.5° bias in yaw that was mentioned in Sec. 6.2.2 and removed from the IMU data. In the absence of the magnetic disturbance that caused this IMU bias to occur in the first place, the IMU should be accurate to 2° during motion and 1° when stationary according to the datasheet. While these solutions are not consistent due to the IMU bias, it is reasonable to expect based on

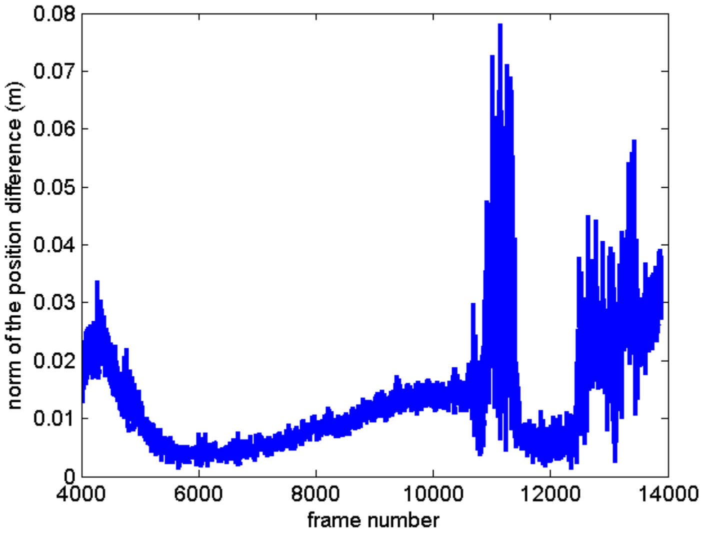

Figure 6.14: This plot shows the norm of the difference between the position of the webcam as estimated by the prototype AR system in coupled CDGPS and INS mode and the calibrated VNS solution from PTAM for the dynamic test.

these results that the two solutions would be consistent if there were no bias in the IMU attitude estimates.

6.3.2 Positioning Results

A trace of the East and North coordinates of the mobile antenna relative to the reference antenna as estimated by the prototype AR system in coupled CDGPS, INS, and VNS mode is shown in Fig. 6.16 for the dynamic test. Only position estimates from after the integer ambiguities were declared converged, which occurred at the same time as in CDGPS mode, are shown in Fig. 6.16. This solution is nearly the same as the coupled CDGPS and INS solution from Fig. 6.9, which was expected based on the consistency of the two solutions demonstrated in Sec. 6.3.1. The VNS corrections to the position estimates were small and are difficult to see at this scale, except for a few places.

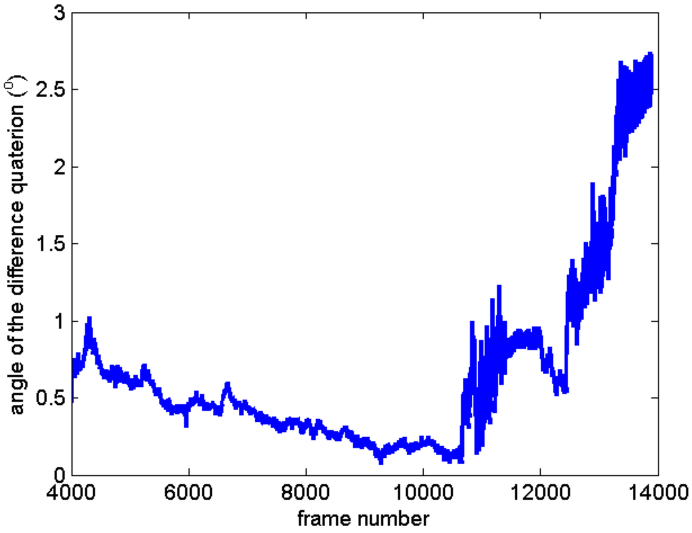

Figure 6.15: This plot shows the rotation angle between the attitude of the webcam as estimated by the prototype AR system in coupled CDGPS and INS mode and the calibrated VNS solution from PTAM for the dynamic test.

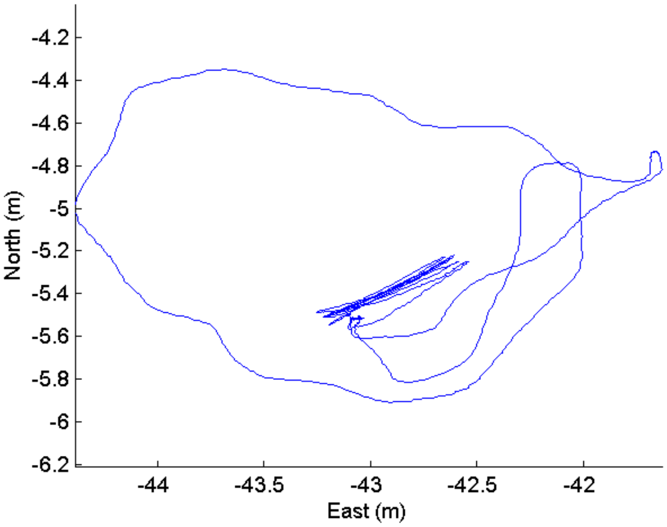

Figure 6.16: This plot shows a trace of the East and North position of the mobile antenna as estimated by the prototype AR system in coupled CDGPS, INS, and VNS mode for the dynamic test from after the integer ambiguities were declared converged. The position estimates are expressed in the ENU frame centered on the reference antenna.

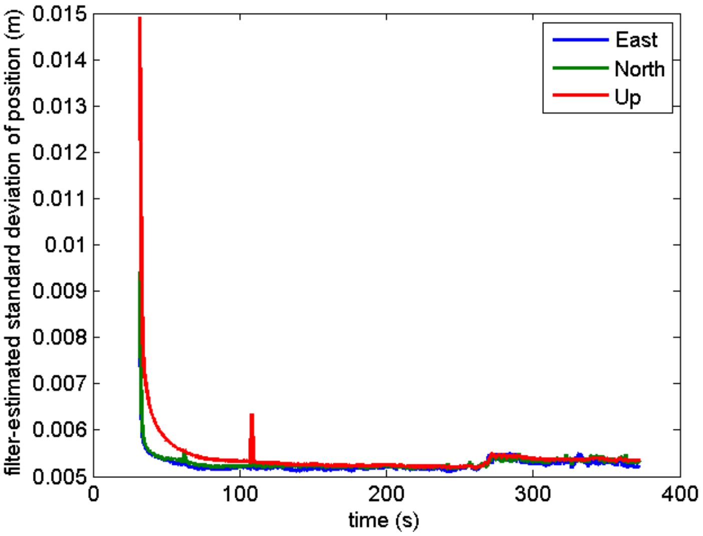

Figure 6.17: This plot shows the standard deviations of the East (blue), North (green), and Up (red) position estimates of the IMU based on the filter covariance estimates from the prototype AR system in coupled CDGPS, INS, and VNS mode for the dynamic test from just before CDGPS measurement updates. The standard deviations are expressed in the ENU frame centered on the reference antenna.

To illustrate the precision of the position estimates, Figs. 6.17 and 6.18 show the standard deviations of the ENU position estimates of the IMU based on the filter covariance estimates from the prototype AR system in coupled CDGPS, INS, and VNS mode from just before and just after CDGPS measurement updates respectively. These standard deviations are significantly smaller than those for the coupled CDGPS and INS mode, shown in Figures 6.10 and 6.11. Note that the covariance on the VNS position estimates was not provided by the VNS, but instead simply chosen to be a diagonal matrix with elements equal to 0.01^2 m^2 based on the consistency results from Sec. 6.3.1.

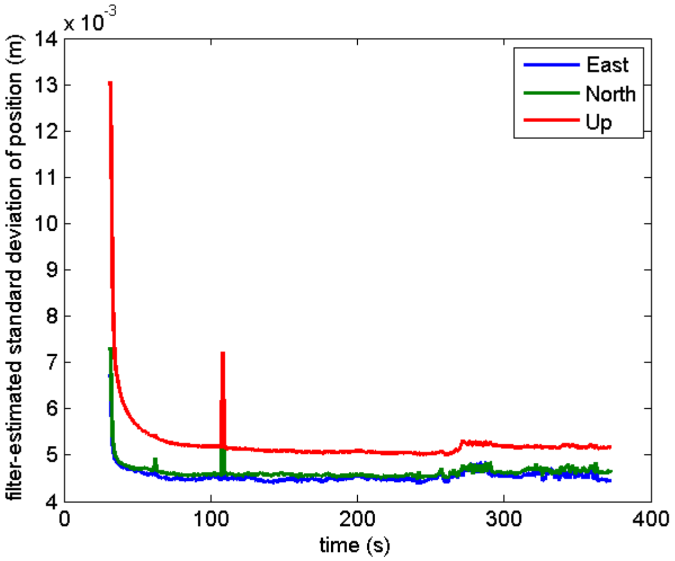

Figure 6.18: This plot shows the standard deviations of the East (blue), North (green), and Up (red) position estimates of the IMU based on the filter covariance estimates from the prototype AR system in coupled CDGPS, INS, and VNS mode for the dynamic test from just after CDGPS measurement updates. The standard deviations are expressed in the ENU frame centered on the reference antenna.

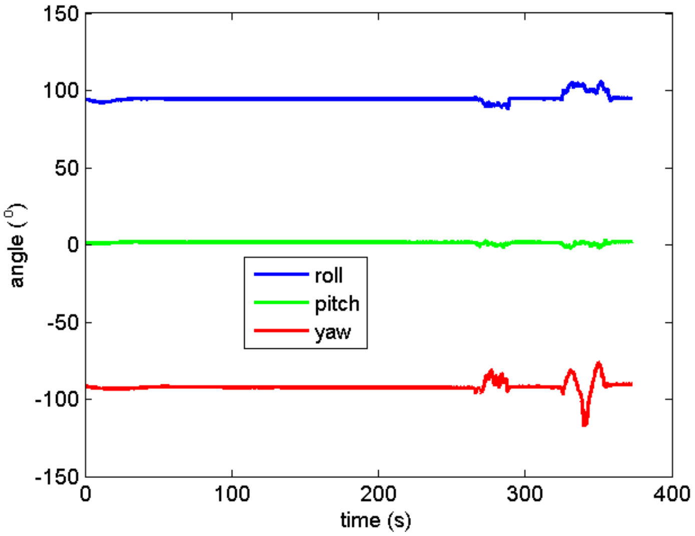

Figure 6.19: This plot shows the attitude estimates from the prototype AR system in coupled CDGPS, INS, and VNS mode for the dynamic test. Attitude is expressed here in the standard yaw-pitch-roll Euler angles for ease of interpretation.

6.3.3 Attitude Results

The attitude estimates, expressed as standard yaw-pitch-roll Euler angle sequences, from the prototype AR system in coupled CDGPS, INS, and VNS mode are shown in Fig. 6.19 for the dynamic test. This solution is nearly the same as the coupled CDGPS and INS solution from Fig. 6.12, which was expected based on the consistency of the two solutions demonstrated in Sec. 6.3.1. One point of difference to note occurs in the yaw estimate near the end of the dataset. It was mentioned in Sec. 6.3.1 that the IMU yaw drifted toward the end of the dataset. The yaw at the end of the dataset should exactly match that during the initial stationary period, since the prototype AR system was returned to the same location at the same orientation for the last 15 to 20 s of the dataset. The inclusion of VNS attitude helped to correct some of this bias. However, this is an unmodeled error in the dataset that could not be completely removed by the filter.

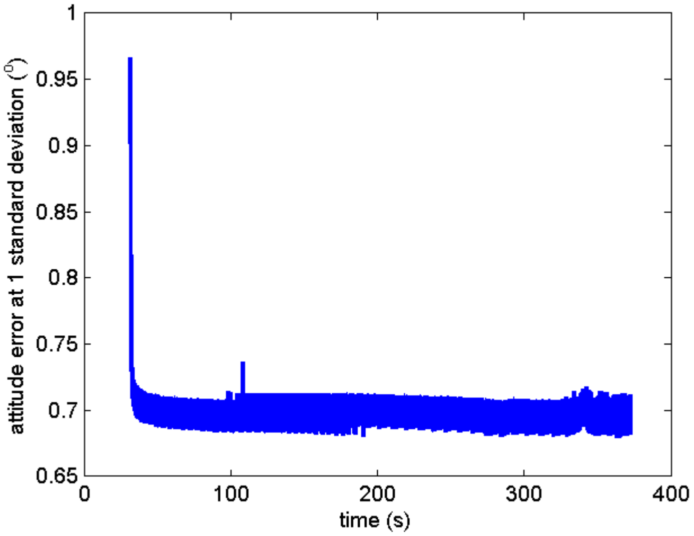

Figure 6.20: This plot shows the standard deviation of the rotation angle between the true attitude and the estimated attitude based on the filter covariance estimates from the prototype AR system in coupled CDGPS, INS, and VNS mode for the dynamic test.

To illustrate the precision of the attitude estimates, Fig. 6.20 shows the expected standard deviation of the rotation angle between the true attitude and the estimated attitude from the prototype AR system in coupled CDGPS, INS, and VNS mode for the dynamic test. This is computed from the filter covariance estimate using Eq. 6.1. This shows that the filter believes the error in its estimate of attitude has a standard deviation of no worse than 0.75° at any time after an initial settling period, which is almost twice as small as that obtained from the prototype AR system in coupled CDGPS and INS mode, as seen in Fig. 6.13. Note that the covariance on the VNS attitude estimates was not provided by the VNS, but instead simply chosen to be a diagonal matrix with elements equal to 0.005^2, which corresponds to a standard deviation of 1°, based on the consistency results from Sec. 6.3.1.

Chapter 7

Conclusions & Future Work

When people think of AR, they imagine a world where both entirely virtual objects and real objects imbued with virtual properties could be used to bring the physical and virtual worlds together. Most existing AR technology has either suffered from poor registration of the virtual objects or been severely limited in application. Some successful AR techniques have been created using visual navigation, but these methods either are only capable of relative navigation, require preparation of the environment, or require a pre-surveyed environment. To reach the ultimate promise of AR, we imagine an ideal AR system that is capable of attaining centimeter-level or better absolute positioning and degree-level or better absolute attitude accuracies in any space, both indoors and out, on a platform that is easy to use and priced reasonably for consumers.

This thesis proposed methods for combining CDGPS, visual SLAM, and inertial measurements to obtain precise and globally-referenced pose estimates of a rigid structure connecting a GPS receiver, a camera, and an IMU. Such a system should be capable of reaching the lofty goals of an ideal AR system. These methods for combining CDGPS, visual SLAM, and inertial measurements include sequential estimators and hybrid batch-sequential estimators. Further analysis is required to identify a single best methodology for this problem, since an optimal solution is computationally infeasible. Prior to defining these estimation methodologies, the observability of absolute attitude based solely on GPS-based position estimates and visual feature measurements was first proven. This eliminates the need for an attitude solution based on magnetometer and accelerometer measurements, which is inaccurate near magnetic disturbances or during long-term sustained accelerations. However, an IMU is still useful for smoothing out dynamics and reduces the drift of the reference frame in GPS-challenged environments.

A prototype AR system was developed as a first step towards the goal of implementing the methods for coupling CDGPS, visual SLAM, and inertial measurements presented in this thesis. This prototype only implemented a loose coupling of CDGPS and visual SLAM, which has difficulty estimating absolute attitude alone because of the need to additionally estimate the similarity transform between ECEF and the arbitrarily-defined frame in which the visual SLAM pose estimates are expressed. Therefore, a full INS was employed by the prototype rather than just inertial measurements. However, the accuracy of both globally-referenced position and attitude are improved over a coupled CDGPS and INS navigation system through the incorporation of visual SLAM in this framework. This prototype demonstrated an upper bound on the precision that such a combination of navigation techniques could attain through a test performed using the prototype AR system. In this test, sub-centimeter-level positioning precision and sub-degree-level obtained precision was attained in a dynamic scenario. This level of precision would enable convincing augmented visuals.

Future work on implementing the proposed methodologies for combining CDGPS, visual SLAM, and inertial measurements and improving the CDGPS algorithm could be performed to bring the system closer to realizing the ultimate promise of AR. First, a comparative performance analysis of the proposed methodologies for combining CDGPS, visual SLAM, and inertial measurements could be performed to determine which methodology is the best based on accuracy and computational expense.

Second, a GPS carrier-phase cycle slip detection and mitigation algorithm could be designed to prevent cycle slips from affecting the CDGPS algorithm.

Appendices

Appendix A

Quaternion Math

This appendix is intended as a brief overview of quaternion math as required by this thesis. More extensive discussions on attitude representations and dynamics can be found in [59, 60].

A.1 Definition of Quaternions

An Irish mathematician named William Rowan Hamilton invented a system of hypercomplex numbers in 1844 called quaternions [61]. This mathematical construct was later adopted for purposes of attitude representation. A quaternion used for attitude representation is composed of a vector and scalar part as follows

$$\mathbf{q} = \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} = \begin{bmatrix} q_0 \\ \mathbf{e} \end{bmatrix} = \begin{bmatrix} \cos\left(\frac{\theta}{2}\right) \\ \hat{e}\sin\left(\frac{\theta}{2}\right) \end{bmatrix} \tag{A.1}$$

where θ and $\hat{e}$ are the angle to rotate through and the axis to rotate about, respectively. The quaternion is also constrained, due to the representation being non-minimal, such that

$$\mathbf{q}^T\mathbf{q} = 1 \tag{A.2}$$

The quaternion representation of attitude is typically preferred for two reasons

- Quaternions avoid the numerical singularity experienced by Euler Angle attitude representations referred to as Gimbal Lock.
- Quaternions typically have better numerical stability as compared to other attitude representations.

A.2 Relation to Direction-Cosine Matrices

It is often more convenient to express attitude as a direction-cosine matrix (DCM) in some calculations because DCMs lend themselves readily to manipulation on computers. The quaternion can be converted into its corresponding direction-cosine matrix as follows

$$R(\mathbf{q}\ ) = \begin{bmatrix} q_0^2+q_1^2-q_2^2-q_3^2 & 2(q_1q_2+q_3q_0) & 2(q_1q_3-q_2q_0) \\ 2(q_1q_2-q_3q_0) & q_0^2-q_1^2+q_2^2-q_3^2 & 2(q_2q_3+q_1q_0) \\ 2(q_1q_3+q_2q_0) & 2(q_2q_3-q_1q_0) & q_0^2-q_1^2-q_2^2+q_3^2 \end{bmatrix} \qquad \text{(A.3)}$$

where $R(\mathbf{q}\ )$ is the direction-cosine matrix. Eq. A.3 can be written more compactly as

$$R(\mathbf{q}\ ) = \left(q_0^2 - ||\mathbf{e}||^2\right) I_{3\times3} + 2\mathbf{e}\mathbf{e}^T - 2q_0[\mathbf{e}\times] \qquad \text{(A.4)}$$

where the cross-product equivalent matrix, $[\mathbf{e}\times]$ is defined as

$$[\mathbf{e}\times] = \begin{bmatrix} 0 & -q_3 & q_2 \\ q_3 & 0 & -q_1 \\ -q_2 & q_1 & 0 \end{bmatrix} \qquad \text{(A.5)}$$

A.3 Quaternion Multiplication

It is conventional to define quaternion multiplication such that

$$R\left(\mathbf{q}_C^B\right) R\left(\mathbf{q}_B^A\right) = R_C^B R_B^A = R_C^A = R\left(\mathbf{q}_C^B \otimes \mathbf{q}_B^A\right) \tag{A.6}$$

where $\mathbf{q}_B^A$ and R_B^A both represent a rotation from the A-frame to the B-frame. This multiplication can be expressed in matrix form as

$$\mathbf{q}_C^B \otimes \mathbf{q}_B^A = \left[\mathbf{q}_C^B\right] \mathbf{q}_B^A \tag{A.7}$$

where $\left[\mathbf{q}_C^B\right]$ is the quaternion left multiplication matrix of $\mathbf{q}_C^B$ defined as

$$[\mathbf{q}] = \begin{bmatrix} q_0 & -q_1 & -q_2 & -q_3 \\ q_1 & q_0 & -q_3 & q_2 \\ q_2 & q_3 & q_0 & -q_1 \\ q_3 & -q_2 & q_1 & q_0 \end{bmatrix} \tag{A.8}$$

The quaternion multiplication can be equivalently expressed as

$$\mathbf{q}_C^B \otimes \mathbf{q}_B^A = \left\{\mathbf{q}_B^A\right\} \mathbf{q}_C^B \tag{A.9}$$

where $\left\{\mathbf{q}_B^A\right\}$ is the quaternion right multiplication matrix of $\mathbf{q}_B^A$ defined as

$$\{\mathbf{q}\} = \begin{bmatrix} q_0 & -q_1 & -q_2 & -q_3 \\ q_1 & q_0 & q_3 & -q_2 \\ q_2 & -q_3 & q_0 & q_1 \\ q_3 & q_2 & -q_1 & q_0 \end{bmatrix} \tag{A.10}$$

A.4 Quaternion Kinematics

The time derivative of the quaternion can be expressed in terms of the angular velocity vector as

$$\dot{\mathbf{q}} = \frac{1}{2}\Omega(\boldsymbol{\omega})\mathbf{q} \tag{A.11}$$

where

$$\Omega(\boldsymbol{\omega}) = \begin{bmatrix} 0 & -\omega_1 & -\omega_2 & -\omega_3 \\ \omega_1 & 0 & \omega_3 & -\omega_2 \\ \omega_2 & -\omega_3 & 0 & \omega_1 \\ \omega_3 & \omega_2 & -\omega_1 & 0 \end{bmatrix} \tag{A.12}$$

A.5 Differential Quaternion

For small rotations, the quaternion can be closely approximated as follows

$$\delta\mathbf{q} = \begin{bmatrix} \cos\left(\frac{\delta\theta}{2}\right) \\ \hat{e}\sin\left(\frac{\delta\theta}{2}\right) \end{bmatrix} \approx \begin{bmatrix} \sqrt{1 - ||\hat{e}\frac{\delta\theta}{2}||^2} \\ \hat{e}\frac{\delta\theta}{2} \end{bmatrix} = \begin{bmatrix} \sqrt{1 - ||\delta\mathbf{e}||^2} \\ \delta\mathbf{e} \end{bmatrix} \tag{A.13}$$

where $\cos\left(\frac{\delta\theta}{2}\right)$ is approximated as $\sqrt{1 - ||\delta\mathbf{e}||^2}$ instead of the typical 1 to comply with the quaternion constraint. This approximation allows for reduction of the quaternion to a minimal three-element representation, $\delta\mathbf{e}$, and is useful for preserving the quaternion constraint in a Kalman Filter, as shown in [48]. During initial convergence of a Kalman Filter, the assumption that $\delta\theta$ is small may be violated and could cause $\sqrt{1 - ||\delta\mathbf{e}||^2}$ to become imaginary. To protect against this scenario, a less accurate form of the differential quaternion is used whenever $||\delta\mathbf{e}||^2 > 1$. This form of the differential quaternion is

$$\delta\mathbf{q} = \frac{1}{\sqrt{1 + ||\delta\mathbf{e}||^2}} \begin{bmatrix} 1 \\ \delta\mathbf{e} \end{bmatrix} \tag{A.14}$$

A useful relation between the differential quaternion and its DCM can be found by linearizing Eq. A.4 about $\delta\mathbf{e} = 0$. This results in the relation

$$R(\delta\mathbf{q}) \approx I_{3\times3} - 2[\delta\mathbf{e}\times] \tag{A.15}$$

Appendix B

Square-Root Extended Kalman Filter

A brief overview of the square-root extended Kalman filter (SREKF) is presented here to provide notational consistency and a brief reference for those familiar with other estimation algorithms. A more in depth discussion of estimation algorithms is included in [38, 62]. Specifically, [62] presents details on square-root information filters for linear systems and [38] discusses estimation algorithms for both linear and non-linear systems primarily in covariance form.

B.1 Propagation Step

The propagation step takes the best estimate of the state, $\hat{\mathbf{X}}(k)$, and state covariance, $P(k)$, at time t_k and evolves these estimates through the state dynamics equations, which may be non-linear, to determine the apriori state, $\bar{\mathbf{X}}(k+1)$, and apriori state covariance, $\bar{P}(k+1)$ at time t_{k+1}. The state dynamics equation can be written in a general form as

$$\dot{\mathbf{X}}(t) = f\left(\mathbf{X}(t), \mathbf{u}(t), t\right) + D(t)\mathbf{v}'(t) \tag{B.1}$$

where $\mathbf{u}(t)$ is the known input to the system, $f\left(\mathbf{X}(t), \mathbf{u}(t), t\right)$ is the dynamics model of the state, $\mathbf{v}(t)$ is the process noise vector, and $D(t)$ maps the process noise to the time derivative of the state vector. The process noise vector is assumed to be zero-mean Gaussian white noise with a covariance given by

$$E[\mathbf{v}'(t)\mathbf{v}'^T(\tau)] = \begin{cases} Q'(t) & ; t = \tau \\ 0 & ; otherwise \end{cases} \tag{B.2}$$

B.1.1 Discretization of Dynamics

For this thesis, the time step, Δt, between propagation steps is small. Therefore, the dynamics model of the state can be approximated as piece-wise constant over the time intervals of the propagation steps:

$$f\left(\mathbf{X}(t), \mathbf{u}(t), t\right) \approx f\left(\hat{\mathbf{X}}(k), \mathbf{u}(k), t_k\right), \quad \forall\ k = 0, 1, \ldots\ \&\ t_k \leq t \leq t_{k+1} \tag{B.3}$$

Under this assumption, the state transition matrix is given by

$$\begin{aligned} F(k) &= e^{A\left(\hat{\mathbf{X}}(k), \mathbf{u}(k), t_k\right)} \\ &\approx I + \Delta t A\left(\hat{\mathbf{X}}(k), \mathbf{u}(k), t_k\right) \end{aligned} \tag{B.4}$$

where

$$A\left(\hat{\mathbf{X}}(k), \mathbf{u}(k), t_k\right) = \left.\frac{\partial f\left(\mathbf{X}(k), \mathbf{u}(k), t_k\right)}{\partial \mathbf{X}}\right|_{\left(\hat{\mathbf{X}}(k), \mathbf{u}(k), t_k\right)} \tag{B.5}$$

The a priori state estimate at time t_{k+1} can also be computed as

$$\bar{\mathbf{X}}(k+1) = E[\mathbf{X}(k+1)] = \hat{\mathbf{X}}(k) + \Delta t f\left(\hat{\mathbf{X}}(k), \mathbf{u}(k), t_k\right) \tag{B.6}$$

The discrete-time process noise covariance matrix is required for propagation of the state covariance. This matrix can be computed as

$$Q(k) = \int_{t_k}^{t_{k+1}} F(k)D(t)Q'(t)D^T(t)F^T(k)dt \tag{B.7}$$

This definition can cause problems with the square-root propagation step if the resulting $Q(k)$ is singular. Specifically, $Q(k)$ becomes singular if some of the states are not affected by process noise. This condition often occurs when constants are included as part of the state. To prevent this issue, the columns and rows of $Q(k)$ containing elements along the primary diagonal which are zero are removed and a matrix $\Gamma(k)$ is defined which maps the reduced discrete-time process noise vector to the state at time t_{k+1}.

The discretized state dynamics equation is thus

$$\mathbf{X}(k+1) = \mathbf{X}(k) + \Delta t f(\mathbf{X}(k), \mathbf{u}(k), t_k) + \Gamma(k)\mathbf{v}(k) \tag{B.8}$$

where $\mathbf{v}(k)$ is the discrete-time process noise with a covariance matrix given by

$$E[\mathbf{v}(k)\mathbf{v}^T(j)] = \begin{cases} Q(k) & ; k = j \\ 0 & ; otherwise \end{cases} \tag{B.9}$$

B.1.2 Formation of Cost Function

The state and process noise can be viewed as the solutions to a set of measurement equations given as

$$\mathbf{z}_X(k) = R_{XX}(k)\mathbf{X}(k) + \mathbf{w}_X(k) \tag{B.10}$$

$$\mathbf{z}_v(k) = 0 = R_{vv}(k)\mathbf{v}(k) + \mathbf{w}_v(k) \tag{B.11}$$

where $\mathbf{w}_X(k)$ and $\mathbf{w}_v(k)$ are zero-mean Gaussian white noise vectors with identity covariance matrices and $R_{XX}(k)$ and $R_{vv}(k)$ are the Cholesky factorizations of $P^{-1}(k)$ and $Q^{-1}(k)$ respectively. These equations can be stacked and expressed as a cost function to be minimized as follows

$$J_a = \frac{1}{2}\left|\left|\begin{bmatrix} R_{vv}(k) & 0 \\ 0 & R_{XX}(k) \end{bmatrix}\begin{bmatrix} \mathbf{v}(k) \\ \mathbf{X}(k) \end{bmatrix} - \begin{bmatrix} 0 \\ \mathbf{z}_X(k) \end{bmatrix}\right|\right|^2 \tag{B.12}$$

B.1.3 Propagating Cost Function Forward in Time

To propagate the state forward in time within the cost function in Eq. B.12, $\mathbf{X}(k+1)$ is substituted for $\mathbf{X}(k)$. This substitution is made by solving the following linearizated version of Eq. B.8 for $\mathbf{X}(k)$

$$\mathbf{X}(k+1) = \hat{\mathbf{X}}(k) + \Delta t f\left(\hat{\mathbf{X}}(k), \mathbf{u}(k), t_k\right) + F(k)\left(\mathbf{X}(k) - \hat{\mathbf{X}}(k)\right) + \Gamma(k)\mathbf{v}(k) \tag{B.13}$$

The resulting cost function is

$$\begin{aligned} J_b = \frac{1}{2}\Bigg|\Bigg| & \begin{bmatrix} R_{vv}(k) & 0 \\ -R_{XX}(k)F^{-1}(k)\Gamma(k) & R_{XX}(k)F^{-1}(k) \end{bmatrix}\begin{bmatrix} \mathbf{v}(k) \\ \mathbf{X}(k+1) \end{bmatrix} \\ & - \begin{bmatrix} 0 \\ \mathbf{z}_X(k) + R_{XX}(k)\left(F^{-1}(k)\left(\hat{\mathbf{X}}(k) + \Delta t f\left(\hat{\mathbf{X}}(k), \mathbf{u}(k), t_k\right)\right) - \hat{\mathbf{X}}(k)\right) \end{bmatrix}\Bigg|\Bigg|^2 \end{aligned} \tag{B.14}$$

To minimize this cost function, a QR factorization is employed to give

$$\begin{aligned} \begin{bmatrix} R_{vv}(k) & 0 \\ -R_{XX}(k)F^{-1}(k)\Gamma(k) & R_{XX}(k)F^{-1}(k) \end{bmatrix} &= \bar{Q}(k)\bar{R}(k) \\ &= \bar{Q}(k)\begin{bmatrix} \bar{R}_{vv}(k) & \bar{R}_{vX}(k) \\ 0 & \bar{R}_{XX}(k+1) \end{bmatrix} \end{aligned} \tag{B.15}$$

Multiplying by $\bar{Q}^T(k)$ from the left inside the norm of the cost function does not change the cost, since $\bar{Q}(k)$ is an orthonormal matrix, and results in a cost function

$$J_c = \frac{1}{2}\left\|\begin{bmatrix} \bar{R}_{vv}(k) & \bar{R}_{vX}(k) \\ 0 & \bar{R}_{XX}(k+1) \end{bmatrix}\begin{bmatrix} \mathbf{v}(k) \\ \mathbf{X}(k+1) \end{bmatrix} - \begin{bmatrix} \bar{\mathbf{z}}_v(k) \\ \bar{\mathbf{z}}_X(k+1) \end{bmatrix}\right\|^2 \tag{B.16}$$

where

$$\begin{aligned} \begin{bmatrix} \bar{\mathbf{z}}_v(k) \\ \bar{\mathbf{z}}_X(k+1) \end{bmatrix} &= \bar{Q}^T(k) \\ \times &\begin{bmatrix} 0 \\ \mathbf{z}_X(k) + R_{XX}(k)\left(F^{-1}(k)\left(\hat{\mathbf{X}}(k) + \Delta t f\left(\hat{\mathbf{X}}(k), \mathbf{u}(k), t_k\right)\right) - \hat{\mathbf{X}}(k)\right) \end{bmatrix} \end{aligned} \tag{B.17}$$

and

$$\bar{P}(k+1) = \left(\bar{R}_{XX}^T(k+1)\bar{R}_{XX}(k+1)\right)^{-1} \tag{B.18}$$

Some filter implementations maintain the process noise equations to help smooth out the past states after more data has been consumed. This operation is referred to as smoothing. Since this filter does not employ a smoother, these equations are no longer used and can be discarded.

B.2 Measurement Update Step

The measurement update step uses information obtained from measurements to correct the a priori state estimate, $\bar{\mathbf{X}}(k)$, to obtain the a posteriori state estimate, $\hat{\mathbf{X}}(k)$. These measurements can be written in the general form

$$\mathbf{z}(k) = h\left(\mathbf{X}(k), k\right) + \mathbf{w}(k) \tag{B.19}$$

where $\mathbf{z}(k)$ is the vector of measurements collected at time t_k, $h\left(\mathbf{X}(k), k\right)$ is the non-linear measurement model, and $\mathbf{w}(k)$ is the measurement noise vector. The measurement noise is assumed to be zero-mean Gaussian white noise with a covariance given by

$$E[\mathbf{w}(k)\mathbf{w}^T(j)] = \begin{cases} R(k) & ; k = j \\ 0 & ; otherwise \end{cases} \tag{B.20}$$

The measurement noise is also assumed to be uncorrelated with the process noise.

In an EKF, the measurement equations from Eq. B.19 are linearized about the a priori state estimate. The linearized measurement model is given by

$$\begin{aligned} h\left(\mathbf{X}(k), k\right) &\approx h\left(\bar{\mathbf{X}}(k), k\right) + \left.\frac{\partial h\left(\mathbf{X}(k), k\right)}{\partial \mathbf{X}(k)}\right|_{\left(\bar{\mathbf{X}}(k), k\right)} \delta\mathbf{X}(k) \\ &= \bar{\mathbf{z}}(k) + H(k)\delta\mathbf{X}(k) \end{aligned} \tag{B.21}$$

where $\delta\mathbf{X}(k)$ is a correction to $\bar{\mathbf{X}}(k)$ typically defined as

$$\delta\mathbf{X}(k) = \hat{\mathbf{X}}(k) - \bar{\mathbf{X}}(k) \tag{B.22}$$

The linearized measurements are thus given by combining Eqs. B.19 and B.21 as

$$\mathbf{z}'(k) = \mathbf{z}(k) - \bar{\mathbf{z}}(k) = H(k)\delta\mathbf{X}(k) + \mathbf{w}(k) \tag{B.23}$$

B.2.1 Formation of Cost Function

In forming the cost function for the measurement update, the linearized measurements from Eq. B.23 are first normalized to obtain

$$\begin{aligned}\tilde{\mathbf{z}}(k) = R_z^{-T}(k)\mathbf{z}'(k) &= R_z^{-T}(k)H(k)\delta\mathbf{X}(k) + R_z^{-T}(k)\mathbf{w}(k) \\ &= \tilde{H}(k)\delta\mathbf{X}(k) + \mathbf{w}_z(k)\end{aligned} \tag{B.24}$$

where $R_z(k)$ is the Cholesky factorization of $R^{-1}(k)$. These normalized measurement equations are stacked on top of the a priori state encoded as measurement equations to obtain the cost function

$$J_d = \frac{1}{2}\left|\left|\begin{bmatrix} \tilde{H}(k) \\ \bar{R}_{XX}(k) \end{bmatrix}\delta\mathbf{X}(k) - \begin{bmatrix} \tilde{\mathbf{z}}(k) \\ \delta\bar{\mathbf{z}}_X(k) \end{bmatrix}\right|\right|^2 \tag{B.25}$$

where

$$\delta\bar{\mathbf{z}}_X(k) = \bar{\mathbf{z}}_X(k) - \bar{R}_{XX}(k)\bar{\mathbf{X}}(k) \tag{B.26}$$

B.2.2 Computation of A Posteriori State Estimate

As with the propagation step, a QR factorization is employed to aid in minimizing the cost function which results in

$$\begin{bmatrix} \tilde{H}(k) \\ \bar{R}_{XX}(k) \end{bmatrix} = \hat{Q}(k)\hat{R}(k) = \hat{Q}(k)\begin{bmatrix} R_{XX}(k) \\ 0 \end{bmatrix} \tag{B.27}$$

Multiplying by $\hat{Q}^T(k)$ from the left inside the norm of the cost function does not change the cost, since $\hat{Q}(k)$ is an orthonormal matrix, and results in a cost function

$$J_e = \frac{1}{2}\left|\left|\begin{bmatrix} R_{XX}(k) \\ 0 \end{bmatrix}\delta\mathbf{X}(k) - \begin{bmatrix} \delta\mathbf{z}_X(k) \\ \mathbf{z}_r(k) \end{bmatrix}\right|\right|^2 \tag{B.28}$$

where

$$\begin{bmatrix} \delta\mathbf{z}_X(k) \\ \mathbf{z}_r(k) \end{bmatrix} = \hat{Q}^T(k) \begin{bmatrix} \tilde{\mathbf{z}}(k) \\ \delta\bar{\mathbf{z}}_X(k) \end{bmatrix} \tag{B.29}$$

The term $\mathbf{z}_r(k)$, typically referred to as the irreducible cost, is a constant that is unaffected by the state. Therefore, it can be removed from the cost function. The cost function is then minimized by choosing the state correction to be

$$\delta\mathbf{X}(k) = R_{XX}^{-1}(k)\delta\mathbf{z}_X(k) \tag{B.30}$$

The state covariance can also be computed, if needed, as

$$P(k) = \left(R_{XX}^T(k)R_{XX}(k)\right)^{-1} \tag{B.31}$$

To prepare for the next propagation step, $\mathbf{z}_X(k)$ is computed using the equation

$$\mathbf{z}_X(k) = \delta\mathbf{z}_X(k) + R_{XX}(k)\bar{\mathbf{X}}(k) \tag{B.32}$$

Appendix C

Integer Least Squares

Integer least squares, also commonly referred to as sphere decoding, has important applications in both multiple-input multiple-output (MIMO) communications systems and GPS. For this thesis, the integer least squares (ILS) problem arises in the CDGPS algorithm when solving for the integer ambiguities on the double-differenced carrier-phase measurements. This discussion is intended as a brief overview of the ILS problem and solution algorithms. Further details on ILS can be found in [54, 63, 64].

C.1 Problem Statement

The ILS problem seeks to minimize a cost function, subject to the constraint that $\mathbf{N}$ be a set of integers, of the form

$$J = \frac{1}{2} \left|\left| R\mathbf{N} - \mathbf{z} \right|\right|^2 \tag{C.1}$$

where R is a square matrix. The matrix R does not need to be square, but a QR factorization could be employed to reduce the information into a set of Mequations, where M is the length of $\mathbf{N}$.

Assuming that R is invertible, a real-valued solution to the problem that minimizes the cost function is given as

$$\mathbf{N}_{real} = R^{-1}\mathbf{z} \tag{C.2}$$

While this solution does minimize the cost, it does not reflect the constraint that $\mathbf{N}$ be a set of integers. One sensible approximate solution would be to round this real valued solution to the nearest integer values for each element in $\mathbf{N}_{real}$ to obtain $\mathbf{N}_{round}$. However, this method can produce very poor results depending of the shape of the cost function.

Another approximate solution called ambiguity bootstrapping is mentioned in [64] which rounds each element as it is solved for in a backward substitution algorithm and uses the integer value for further steps. This approach performs much better than simply rounding the real-valued solution, but can still settle far from the optimal solution.

C.2 Solution Algorithms

The optimal solution to the ILS problem must be determined through a search algorithm that tests individual lattice points to find the integer set N_{opt} that minimizes Eq. C.1. However, this search can be a computationally intensive process because it requires that many lattice points be tested to ensure that the optimal integer set is found. Theoretically, it has been shown that this problem is NP-hard [63].

Several different algorithms have been created to address the problem of reducing the number of lattice points that must be tested. These algorithms involve three primary steps as follows

reduction step: The reduction step, sometimes referred to as decorrelation, seeks to simplify the ILS problem through matrix factorizations of R. These factorizations typically result in a lattice with bases that are nearly orthogonal with near unity magnitude. For a lattice with orthogonal bases, $\mathbf{N}_{round}$ would be the optimal solution. Having nearly orthogonal bases makes the search step more efficient because the approximate solution is closer in cost to the optimal solution.

approximate solution: An approximate solution is determined using a computationally efficient, but sub-optimal, procedure. This could be as simple as rounding the real valued solution, but better approximations exist. The cost of this approximate solution is computed from Eq. C.1 and is used to guide the search algorithm.

search step: A search tree is created based on all lattice points inside the cost ellipse at the cost of the approximate solution. The optimal solution is guaranteed to have the same or lower cost as compared to the approximate solution, since the approximate solution is also an admissible solution. From this discussion, it can easily be seen that a lower cost approximate solution results in fewer lattice points within the cost ellipse and thus better computational efficiency of the search step.

The two most prominent algorithms employed to solve ILS are the least-squares ambiguity decorrelation adjustment (LAMBDA) method and the Lenstra, Lenstra, Jr., and Lovasz (LLL) method. Psiaki and Mohiuddin present an implementation of the LAMBDA method in [64] which operates on the square-root form of the ILS problem, as shown in Eq. C.1. An implementation of LLL by Hassibi and Boyd can be found in [63] alongside easily-computable, tight upper and lower bounds on the probability that the optimal integer set is correct.

Both algorithms perform well, but the LLL method was chosen for use in this thesis due to the availability of these easily computable probability bounds. In particular, a modified version of MILES by Chang et al. [54] was used for this thesis.

C.3 Solution Verification

For application of ILS to CDGPS, it is important to know when the integer ambiguities have been correctly determined. The actual probability that the integer set is correct can be computed by integrating a multivariate Gaussian distribution over a Voronoi cell. A Voronoi cell is defined by the set of all points closest to a single integer set. Numerical integration over a Voronoi cell is incredibly computationally intensive and thus infeasible for real-time applications [63].

Hassibi and Boyd presented upper and lower bounds on the probability that the optimal integer set is correct in [63] that utilize the LLL reduced version of the matrix R from Eq. C.1. These probability bounds become tight as the probability approaches 1, which is a highly desirable quality. The upper bound probability is based on an integral of the probability distribution over a Euclidean ball that contains the entirety of the Voronoi cell. This upper bound can be computed as

$$P_{up} = F_{\chi^2}\left(\left(\left|det(G)\right|/\alpha_M\right)^{2/M}; M\right) \tag{C.3}$$

where $F_{\chi^2}(\cdot; M)$ is the χ^2 cumulative distribution function with M degrees of freedom, G is the LLL reduced version of R, and α_M is given by

$$\alpha_M = \frac{\pi^{M/2}}{\Gamma\left(M/2+1\right)} \tag{C.4}$$

where $\Gamma(\cdot)$ is the Gamma function. The lower bound probability is based on an integral of the probability distribution over a Euclidean ball entirely inside the Voronoi cell. This lower bound can be computed as

$$P_{low} = F_{\chi^2}\left(\frac{d_{min}^2}{4}; M\right) \tag{C.5}$$

where d_{min} is the shortest distance between lattice points. Computing d_{min} is NP-hard, but a tight lower bound on d_{min} can be computed from the Gram-Schmidt orthogonalization of G as

$$d_{min} \geq min\left(||\mathbf{g}_1^*||, ||\mathbf{g}_2^*||, \ldots, ||\mathbf{g}_M^*||\right) \tag{C.6}$$

where $\mathbf{g}_i^*$ are the columns of the Gram-Schmidt orthogonalization of G.

Bibliography

[1] "Word lens," Quest Visual, 2013, http://questvisual.com/us/.

[2] V. Technology, "Start walk," http://vitotechnology.com/star-walk.html, 2012, [Online; accessed 28-September-2012].

[3] S. Feiner, B. MacIntyre, T. Höllerer, and A. Webster, "A touring machine: Prototyping 3d mobile augmented reality systems for exploring the urban environment," *Personal and Ubiquitous Computing*, vol. 1, no. 4, pp. 208–217, 1997.

[4] G. Roberts, A. Evans, A. Dodson, B. Denby, S. Cooper, R. Hollands *et al.*, "The use of augmented reality, gps and ins for subsurface data visualization," in *FIG XXII International Congress*, 2002, pp. 1–12.

[5] P. Wellner, W. Mackay, and R. Gold, "Back to the real world," in *Communications of the ACM*, vol. 36, no. 7. ACM, 1993, pp. 24–26.

[6] R. Azuma *et al.*, "A survey of augmented reality," *Presence-Teleoperators and Virtual Environments*, vol. 6, no. 4, pp. 355–385, 1997.

[7] "Glass: What it does," Google, Mar. 2013, http://www.google.com/glass/start/what-it-does/.

[8] D. H. Shin and P. S. Dunson, "Identification of application areas for augmented reality in industrial construction based on technology suitability," *Automation in Construction*, vol. 17, pp. 882–894, Feb. 2008.

[9] Sportsvision, "1st and ten line system," http://www.sportvision.com/foot-1st-and-ten-line-system.html, 2012, [Online; accessed 27-September-2012].

[10] Lego, "Lego augmented reality kiosks heading to shops worldwide," 2010.

[11] S. Perry, "Wikitude: Android app with augmented reality: Mind blowing," Oct. 2008, http://digital-lifestyles.info/2008/10/23/wikitude-android-app-with-augmented-reality-mind-blowing/.

[12] "Layar," http://www.layar.com/, [Online; accessed 14-April-2013].

[13] B. Huang and Y. Gao, "Indoor navigation with iPhone/iPad: Floor plan-based monocular vision navigation," in *Proceedings of the ION GNSS Meeting*. Nashville, TN: Institute of Navigation, Sept. 2012.

[14] D. Zachariah and M. Jansson, "Fusing visual tags and inertial information for indoor navigation," in *Proceedings of the IEEE/ION PLANS Meeting*. Myrtle Beach, SC: IEEE/Institute of Navigation, April 2012.

[15] N. Snavely, S. M. Seitz, and R. Szeliski, "Photo tourism: Exploring photo collections in 3d," *ACM Transactions on Graphics*, vol. 25, no. 3, pp. 835–846, 2006.

[16] B. A. y Arcas, "Blaise aguera y arcas demos augmented-reality maps," TED, Feb. 2010, http://www.ted.com/talks/blaise_aguera.html.

[17] A. H. Bahzadan and V. R. Kamat, "Georeferenced registration of construction graphics in mobile outdoor augmented reality," *Journal of Computing in Civil Engineering*, vol. 21, no. 4, July 2007.

[18] A. H. Behzadan, B. W. Timm, and V. R. Kamat, "General-purpose modular hardware and software framework for mobile outdoor augmented reality applications in engineering," *Advanced Engineering Informatics*, vol. 22, pp. 90–105, 2008.

[19] G. Roberts, X. Meng, A. Taha, and J.-P. Montillet, "The location and positioning of buried pipes and cables in built up areas," in *XXIII FIG Congress*, Munich, Germany, Oct. 2006.

[20] G. Schall, E. Mendez, E. Kruijff, E. Veas, S. Junghanns, B. Reitinger, and D. Schmalstieg, "Handheld augmented reality for underground infrastructure visualization," *Personal and Ubiquitous Computing*, vol. 13, no. 4, pp. 281–291, 2009.

[21] G. Schall, D. Wagner, G. Reitmayr, E. Taichmann, M. Wieser, D. Schmalstieg, and B. Hofmann-Wellenhof, "Global pose estimation using multi-sensor fusion for outdoor augmented reality," in *IEEE International Symposium on Mixed and Augmented Reality*. Orlando, FL: IEEE, Oct. 2009.

[22] G. Schall, S. Zollmann, and G. Reitmayr, "Smart vidente: Advances in mobile augmented reality for interactive visualization of underground infrastructure," *Pers Ubiquit Comput*, July 2012.

[23] A. I. Mourikis and S. I. Roumeliotis, "A multi-state constraint kalman filter for vision-aided inertial navigation," in *Robotics and Automation, 2007 IEEE International Conference on*. IEEE, 2007, pp. 3565–3572.

[24] J. Rydell and E. Emilsson, "CHAMELEON: Visual-inertial indoor navigation," in *Proceedings of the IEEE/ION PLANS Meeting*. Myrtle Beach, SC: IEEE / Institute of Navigation, April 2012.

[25] A. Soloviev, J. Touma, T. Klausutis, M. Miller, A. Rutkowski, and K. Fontaine, "Integrated multi-aperture sensor and navigation fusion," in *Proceedings of the ION GNSS Meeting*. Savannah, GA: Institute of Navigation, Sept. 2009.

[26] L. Kneip, S. Weiss, and R. Siegwart, "Deterministic initialization of metric state estimation filters for loosely-coupled monocular vision-inertial systems," in *Proceedings of the IEEE Conference on Intelligent Robots and Systems*. San Francisco, CA: IEEE, Sept. 2011.

[27] R. Brockers, S. Susca, D. Zhu, and L. Matthies, "Fully self-contained vision-aided navigation and landing of a micro air vehicle independent from external sensor inputs," in *Proceedings of Unmanned Systems Technology XIV*. Bellingham, WA: SPIE, 2012.

[28] G. Nuetzi, S. Weiss, D. Scaramuzza, and R. Siegwart, "Fusion of IMU and vision for absolute scale estimation in monocular SLAM," *Journal of Intelligent & Robotic Systems*, vol. 61, no. 1, pp. 287–299, Jan. 2011.

[29] S. Weiss and R. Siegwart, "Real-time metric state estimation for modular vision-inertial systems," in *Proceedings of the IEEE Conference on Robotics and Automation*. IEEE, May 2011.

[30] C. N. Taylor, "An analysis of observability-constrained kalman filtering for vision-aided navigation," in *Proceedings of the IEEE/ION PLANS Meeting*. Myrtle Beach, SC: IEEE/Institute of Navigation, April 2012, pp. 1240–1246.

[31] D. H. Won, S. Sung, and Y. J. Lee, "Ukf based vision aided navigation system with low grade imu," in *Proceedings of the International Conference on Control, Automation and Systems*, Oct. 2010.

[32] A. Soloviev and D. Venable, “Integration of GPS and vision measurements for navigation in GPS challenged environments,” in *Proceedings of the IEEE/ION PLANS Meeting.* IEEE/Institute of Navigation, May 2010, pp. 826–833.

[33] J. Wang, M. Garratt, A. Lambert, J. J. Wang, S. Han, and D. Sinclair, “Integration of gps/ins/vision sensors to navigate unmanned aerial vehicles,” *The International Archives of the Photogrammetry, Remote Sensing, and Spatial Information Sciences*, vol. 37, no. B1, pp. 963–969, 2008.

[34] J. J. Koenderink, A. J. Van Doorn *et al.*, “Affine structure from motion,” *JOSA A*, vol. 8, no. 2, pp. 377–385, 1991.

[35] S. Ullman, *Interpretation of Visual Motion.* Cambridge, MA: The MIT Press, 1979.

[36] B. K. Horn, “Closed-form solution of absolute orientation using unit quaternions,” *JOSA A*, vol. 4, no. 4, pp. 629–642, 1987.

[37] H. Strasdat, J. Montiel, and A. J. Davison, “Visual slam: Why filter?” *Image and Vision Computing*, 2012.

[38] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation.* New York: John Wiley and Sons, 2001.

[39] R. E. Kalman, “A new approach to linear filtering and prediction problems,” *Journal of Basic Engineering*, vol. 82, pp. 35–45, 1960.

[40] R. E. Kalman and R. S. Bucy, “New results in linear filtering and prediction theory,” *Journal of Basic Engineering*, vol. 83, pp. 95–108, Mar. 1961.

[41] M. Psiaki, “Backward-smoothing extended kalman filter,” *Journal of guidance, control, and dynamics*, vol. 28, no. 5, pp. 885–894, 2005.

[42] S. J. Julier and J. K. Uhlmann, “Unscented filtering and nonlinear estimation,” *Proceedings of the IEEE*, vol. 93, no. 3, pp. 401–422, Mar. 2004.

[43] J. S. Liu and R. Chen, “Sequential monte carlo methods for dynamic systems,” *Journal of The American Statistical Association*, vol. 93, no. 443, pp. 1032–1044, 1998.

[44] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision.* Cambridge Univ Press, 2000, vol. 2.

[45] G. Klein and D. Murray, “Parallel tracking and mapping for small AR workspaces,” in *6th IEEE and ACM International Symposium on Mixed and Augmented Reality.* IEEE, 2007, pp. 225–234.

[46] B. Triggs, P. McLauchlan, R. Hartley, and A. Fitzgibbon, “Bundle adjustmenta modern synthesis,” *Vision algorithms: theory and practice*, pp. 153–177, 2000.

[47] D. P. Shepard, K. M. Pesyna, and T. Humphreys, “Precise augmented reality enabled by carrier-phase differential GPS,” in *Proceedings of the ION GNSS Meeting.* Nashville, Tennessee: Institute of Navigation, 2012.

[48] T. E. Humphreys, “Attitude determination for small satellites with modest pointing constraints,” in *Proc. 2002 AIAA/USU Small Satellite Conference*, Logan, Utah, 2002.

[49] P. Misra and P. Enge, *Global Positioning System: Signals, Measurements, and Performance.* Lincoln, Massachusetts: Ganga-Jumana Press, 2006.

[50] S. Mohiuddin and M. Psiaki, "Carrier-phase differential Global Positioning System navigation filter for high-altitude spacecraft," *Journal of guidance, control, and dynamics*, vol. 31, no. 4, pp. 801–814, 2008.

[51] S. Mohiuddin and M. L. Psiaki, "High-altitude satellite relative navigation using carrier-phase differential global positioning system techniques," *Journal of Guidance, Control, and Dynamics*, vol. 30, no. 5, pp. 1628–1639, Sept.-Oct. 2007.

[52] J. Farrell, T. Givargis, and M. Barth, "Real-time differential carrier phase GPS-aided INS," *Control Systems Technology, IEEE Transactions on*, vol. 8, no. 4, pp. 709–721, 2000.

[53] W. S. Flenniken IV, J. H. Wall, and D. M. Bevly, "Characterization of various imu error sources and the effect on navigation performance," in *Proceedings of the ION ITM*. Long Beach, California: Institute of Navigation, 2005.

[54] X. W. Chang, X. Xie, and T. Zhou, *MILES: MATLAB package for solving Mixed Integer LEast Squares problems*, 2nd ed., http://www.cs.mcgill.ca/chang/software.php, Oct. 2011.

[55] B. O'Hanlon, M. Psiaki, S. Powell, J. Bhatti, T. E. Humphreys, G. Crowley, and G. Bust, "CASES: A smart, compact GPS software receiver for space weather monitoring," in *Proceedings of the ION GNSS Meeting*. Portland, Oregon: Institute of Navigation, 2011.

[56] "GPS antennas for avionics, ground, and marine applications," Antcom, 2008, http://www.antcom.com/documents/catalogs/L1L2GPSAntennas.pdf.

[57] T. E. Humphreys, M. L. Psiaki, P. M. Kitner, and B. M. Ledvina, "GNSS receiver implementation on a DSP: Status, challenges, and prospects," in *Proceedings of the ION GNSS Meeting.* Fort Worth, TX: The Institute of Navigation, 2006.

[58] T. E. Humphreys, J. Bhatti, T. Pany, B. Ledvina, and B. O'Hanlon, "Exploiting multicore technology in software-defined GNSS receivers," in *Proceedings of the ION GNSS Meeting.* Savannah, GA: Institute of Navigation, 2009.

[59] V. L. Piscane and R. C. Moore, *Fundamentals of Space Systems.* Oxford, UK: Oxford University Press, 1994.

[60] H. D. Curtis, *Orbital Mechanics for Engineering Students*, 2nd ed. Burlington, MA: Elsevier, 2009.

[61] W. R. Hamilton, "On quaternions, or on a new system of imaginaries in algebra," *Philosophical Magazine*, vol. 25, no. 3, pp. 489–495, 1844.

[62] G. J. Bierman, *Factorization Methods for Discrete Sequential Estimation.* New York: Academic Press, 1977.

[63] A. Hassibi and S. Boyd, "Integer parameter estimation in linear models with applications to gps," *Signal Processing, IEEE Transactions on*, vol. 46, no. 11, pp. 2938–2952, 1998.

[64] M. Psiaki and S. Mohiuddin, "Global positioning system integer ambiguity resolution using factorized least-squares techniques," *Journal of Guidance, Control, and Dynamics*, vol. 30, no. 2, pp. 346–356, March-April 2007.