2007

Automatic geo-referencing by integrating camera vision and inertial measurements

Duminda I. B Randeniya
University of South Florida

Automatic Geo-Referencing by Integrating Camera Vision and Inertial Measurements

by

Duminda I. B. Randeniya

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
Department of Civil and Environmental Engineering
College of Engineering
University of South Florida

Co-Major Professor: Manjriker Gunaratne, Ph.D.
Co-Major Professor: Sudeep Sarkar, Ph.D.
Ram Pendyala, Ph.D.
Auroop Ganguly, Ph.D.
George Yanev, Ph.D.

Date of Approval:
June 22, 2007

Keywords: Multi-Sensor Fusion, Vision/INS Integration, Intelligent Vehicular Systems,
Computer Vision, Inertial Navigation

DEDICATION

This work is dedicated to my parents Tissa and Hema Randeniya and to my two sisters Dinishi and Sajini. Without their support and encouragement I would never achieve this.

ACKNOWLEDGEMENTS

I would like to thank my co-advisors, Dr. Manjriker Gunaratne and Dr. Sudeep Sarkar for providing me with the opportunity to work on this project. Your guidance, patience and mentorship are greatly appreciated. I truly could not have asked for a better people to have worked under. Also, the Florida Department of Transportation should be thanked for supporting my research efforts through a research grant to the University of South Florida.

Additionally, I would like to extend my thanks to the members of my examining committee, Dr. Auroop Ganguly, Dr. Ram Pendyala, Dr. George Yanev and Dr. John Lu for all the support and courage given to me through out my stay at USF. Also I thank Dr. Elaine Chang for her support. Farther, I would be very grateful to my colleagues especially M. Jeyishanker, K. Jeyishanker and L. Fuentes for helping me at different occasions surveying the roadways under scorching summer sun.

I would also like to express my sincere appreciation to the State Materials Office personnel in Gainesville for their continuous support throughout the course of the project and allowing me to collect vital data for my dissertation. In specific, the help of Mr. Abdenour Nazef, Mr. Stacy Scott and Mr. Glenn Salvo is acknowledged.

Last but not the least I would like to give my warmest gratitude to my parents and two sisters for their unconditional support and love. I wouldn't have achieved this without their guidance and support.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

AUTOMATIC GEO-REFERENCING BY INTEGRATING CAMERA VISION AND INERTIAL MEASUREMENTS

D. I. B. RANDENIYA

ABSTRACT

Importance of an alternative sensor system to an inertial measurement unit (IMU) is essential for intelligent land navigation systems when the vehicle travels in a GPS deprived environment. The sensor system that has to be used in updating the IMU for a reliable navigation solution has to be a passive sensor system which does not depend on any outside signal. This dissertation presents the results of an effort where position and orientation data from vision and inertial sensors are integrated. Information from a sequence of images captured by a monocular camera attached to a survey vehicle at a maximum frequency of 3 frames per second was used in upgrading the inertial system installed in the same vehicle for its inherent error accumulation. Specifically, the rotations and translations estimated from point correspondences tracked through a sequence of images were used in the integration. However, for such an effort, two types of tasks need to be performed. The first task is the calibration to estimate the intrinsic properties of the vision sensors (cameras), such as the focal length and lens distortion parameters and determination of the transformation between the camera and the inertial systems. Calibration of a two sensor system under indoor conditions does not provide an appropriate and practical transformation for use in outdoor maneuvers due to invariable differences between outdoor and indoor conditions. Also, use of custom calibration

objects in outdoor operational conditions is not feasible due to larger field of view that requires relatively large calibration object sizes. Hence calibration becomes one of the critical issues particularly if the integrated system is used in Intelligent Transportation Systems applications. In order to successfully estimate the rotations and translations from vision system the calibration has to be performed prior to the integration process.

The second task is the effective fusion of inertial and vision sensor systems. The automated algorithm that identifies point correspondences in images enables its use in real-time autonomous driving maneuvers. In order to verify the accuracy of the established correspondences, independent constraints such as *epipolar lines* and *correspondence flow directions* were used. Also a pre-filter was utilized to smoothen out the noise associated with the vision sensor (camera) measurements. A novel approach was used to obtain the geodetic coordinates, i.e. latitude, longitude and altitude, from the normalized translations determined from the vision sensor. Finally, the position locations based on the vision sensor was integrated with those of the inertial system in a decentralized format using a Kalman filter. The vision/inertial integrated position estimates are successfully compared with those from 1) inertial/GPS system output and 2) actual survey performed on the same roadway. This comparison demonstrates that vision can in fact be used successfully to supplement the inertial measurements during potential GPS outages. The derived intrinsic properties and the transformation between individual sensors are also verified during two separate test runs on an actual roadway section.

CHAPTER 1 – INTRODUCTION

1.1 Background

Error growth in the measurements of inertial systems is a major issue that limits the accuracy of inertial navigational systems. However, due to the high accuracy associated with inertial systems in short term applications, many techniques such as Differential Global Positioning Systems (DGPS), camera (vision) sensors etc. have been experimented by researchers to be used in conjunction with inertial systems and overcome the error growth in long-term applications. A breakthrough in Intelligent Transportation Systems (ITS) and Mobile Mapping Technology (Cramer, 2005) was made when the Global Positioning System (GPS) and the Inertial Navigation System (INS) was successfully integrated using a Kalman filter. This integration has been achieved in different integration architectures (Wei and Schwarz, 1990) and applied widely since, it addresses the above mentioned error accumulation in Inertial Measurement Unit (IMU). With the improved accuracies achieved from the differential correction of GPS signal and elimination of the *Selective Availability*, a variety of promising research endeavors in INS/GPS integration technology have sprung up. Yet, a vulnerability of the INS/GPS system lies in the fact that it solely relies on the GPS signal to correct the position and attitude estimation of the vehicle in the longer run. Therefore, INS/GPS integration system fails to provide a sufficiently accurate continuous output in

the presence of GPS outages either in urban canyons or locations where communication is hindered (Feng and Law, 2002).

To overcome this issue in real-time navigation, researchers have attempted various alternatives; development of highway infrastructure and Automated Highway Systems (AHS) with embedded magnetic markers (Farrell and Barth, 2002) and vehicles with devices that can locate these markers. Although the combination of AHS with ITS would provide more reliable measurements, improvement of infrastructure is costly. Hence, alternative means of position location with wireless or mobile phone networks (Zhao, 2000) that preclude the need for upgrading the highway infrastructure, have also been explored. At the same time, use of mobile phone networks or any signal-based locating systems have the disadvantage of possible signal outages and blockages which cause system failure. Even when the signal is available the vehicle location estimation accuracy is limited since mobile network base stations have a 50-100m error band. Therefore, the use of primary *vehicle-based*, and external *signal-independent* systems are vital to reliable position and orientation estimation in all types of terrain.

On the other hand, due to the advances in computer vision, potentially promising studies that involve vision sensing are being carried out in the areas of Intelligent Transport Systems (ITS) and Automatic Highway Systems (AHS). The most common applications of vision systems include detection of obstacles such as pedestrians crossing a roadway, sudden movement of other vehicles (Kumar et al., 2005; Sun et al., 2004; Franke and Heinrich, 2002) vehicle maneuvers such as lane changing (Dellaert and Thorpe, 1997;

Sotelo et al., 2004; Li and Leung, 2004) and utilizing vision sensors to aid autonomous navigation without fusing the vision sensor to other sensor systems that measure absolute position such as GPS (Bertozzi et al. 1998, Bertozzi et al. 2000, McCall et al. 2006). In detecting the moving objects in these applications researchers first track the surrounding area for moving objects or vehicles by processing images and using computer vision techniques. Once a moving object is located, the movement of the object is tracked using a tracking algorithm, in order to avoid collision.

In addition, researchers (Sotelo et al. 2004, Bertozzi et al. 1998, Bertozzi et al. 2000, McCall et al. 2006) used vision measurements to track the lane markings and other vehicles traveling in the vicinity. For this purpose, they used image processing techniques such as segmentation, with a second order polynomial function, to approximate the road geometry. This development made it possible to identify the position of the vehicle with respect to the pavement boundaries. Also these authors used GPS measurements combined with the positions of the vehicle obtained from vision system with respect to the tracked lane markings in navigating the vehicle. However, these methods do not use inertial navigation system measurements in estimating the position of the vehicle.

Also researchers have experimented combining inertial sensors with vision sensors, in navigation. Most common application of this type of combination has been in an indoor environment (Diel, 2005, Foxlin 2003) where a vision sensor is used to identify a landmark and perform the navigation task accordingly. In such situations artificial landmarks have been placed for the vision sensor to obtain feature correspondences

conveniently (Foxlin 2003, Hu 2004). Obviously, one cannot expect the same indoor algorithms to perform with the same degree of accuracy in an uncontrolled environment. Therefore, it is essential that algorithms be designed for uncontrolled (outdoor) environments that also incorporate tools to filter out erroneous correspondences. For a vision algorithm to produce accurate rotations and translations, either the selected feature correspondences or the camera must be stationary (Faugeras 1996). However, this condition can be violated easily when correspondence techniques developed for indoor conditions are extended to outdoors. This is due to the possibility of uncontrolled selection of correspondences from moving objects by the camera which itself moves with the vehicle.

As more and more studies are being conducted in integrating inertial and vision sensors, one of the crucial issues faced by researchers is the appropriate calibration of the inertial sensor with the vision sensor. Whether the application that the integrated system is intended for use is indoors (e.g. indoor robot navigation) or outdoors (e.g. vision aided inertial navigation of a vehicle) accurate calibration becomes critical. This is because, for successful integration of vision and inertial data, the mathematical relationship (transformation) between these two measurement systems must be determined accurately. In spite of its vitality in implementing the integration mechanism, only limited literature is available on reports of successful calibration. The initial groundwork to estimate the relative position and orientation (*pose*) of two independent systems with separate coordinate frames was laid, in the form of a mathematical formulation, in (Horn, 1987). Horn (Horn, 1987) used unit *quaternions* in obtaining a closed-form solution for the

absolute orientation between two different coordinate frames by using three non-collinear points. In extending this work to the calibration of inertial and vision sensor integrated systems, (Alves et al., 2003) describe a procedure that can be used in indoor settings where both inertial and vision sensors are fixed to a pendulum. The approach used in (Alves et al., 2003) is applicable in determining the relative *pose* between the two sensors when both of them sense vertical quantities, i.e. when the inertial sensor measures accelerations due to gravity and the vision sensor captures images with predominantly vertical edges. This idea was further extended in (Lang and Pinz, 2005) to incorporate measurements in any desired direction without constraining the measurements to the vertical direction. The relevant calibration was also performed in an indoor setting in a controlled environment and simulated using MATLAB. In addition, (Foxlin et al., 2003) used a specially constructed calibration rig and a stand to estimate the relative *pose* between inertial and vision sensors.

Roumeliotis et al. (Roumeliotis 2002) designed a vision inertial fusion system for use in landing a space vehicle using aerial photographs and an inertial measuring unit (IMU). This system was designed using an indirect Kalman filter, which incorporates the errors in the estimated position estimation, for the input of defined *pose* from camera and IMU systems. However, the fusion was performed on the relative *pose* estimated from the two sensor systems and due to this reason a much simpler inertial navigation model was used compared to that of an absolute *pose* estimation system. Testing was performed on a gantry system designed in the laboratory. Chen et al. (Chen 2004) investigated the estimation of a structure of a scene and motion of the camera by integrating a camera

system and an inertial system. However, the main task of this fusion was to estimate the accurate and robust *pose* of the camera. Foxlin et al. (Foxlin 2003) used inertial vision integration strategy to develop a miniature self-tracker which uses artificial fiducials. Fusion was performed using a bank of Kalman filters designed for acquisition, tracking and hybrid tracking of these fiducials. The IMU data was used to predict the vicinity of the fiducials in the next image. On the other hand, You et al. (You 2001) developed an integration system that could be used in Augmented Reality (AR) applications. This system used a vision sensor in estimating the relative position whereas the rotation was estimated using gyroscopes. No accelerometers were used in the fusion. Dial et al. (Dial 2005) used an IMU and a vision integration system in navigating a robot under indoor conditions. Gyroscopes were used to get the rotation of the cameras and the main target of the fusion was to interpret the visual measurements. Finally, Huster et al. (Huster 2003) used the vision inertial fusion to position an autonomous underwater vehicle (AUV) relative to a fixed landmark. Only one landmark was used in this process making it impossible to estimate the *pose* of the AUV using a camera so that the IMU system was used to fulfill this task.

1.2 Scope of Work

Table 1 illustrates a summary of researches that use IMU/Vision integrated systems. Although most of the work on IMU/Vision integration work has been performed in augmented reality, yet a few other works involves actual conditions. Furthermore, most of the systems have been tested and used in indoor and controlled environments with a few only tested in outdoor and uncontrolled settings. The above are some of the critical problems one has to consider when designing an IMU/Vision system for land navigation

system. Hence the contents of Table1 are of immense significance in illustrating the addressing of these vital factors in IMU/Vision fusion systems and highlighting the work to be performed.

Table 1: Current research performed in IMU/Vision integration (x - no and √ - yes).

Reference	Augmented Realty	IMU for camera *pose*	*pose* from camera	INS vision fusion	Filter	IMU nav. equations	Outdoor testing	Camera speed	Control data	Results compared
Roumeliotis et al. (2002)	x	x	√	√	KF	x	x	-	x	-
Chen et al. (2004)	x	x	-	√	KF	√	x	-	√	-
Foxlin et al. (2003)	√	x	-	√	KF	√	x	24 Hz	√	real-data
You et al. (2001)	√	x	√	√	EKF	√	x	30Hz	√	-
Dial et al. (2005)	x	Rota only	T only	√	MLKF	√	x	10 Hz	x	Raw (Un-aid) data
Huster et al. (2003)	x	√	x	√	EKF		√	-	x	-
Hu et al. (2004)	x	x	√	√	PMM	x	√	-	x	Back projection
This dissertation	x	x	√	√	KF	√	√	3 Hz	x	Real & Survey data

The abbreviations KF, EKF, MLKF and PMM stand for Kalman filter, Extended Kalman filter, Modified Linear Kalman filter and Parameterized model matching respectively.

The fusion approach presented in this dissertation differs from the above mentioned work in many respects. One of the key differences is that the vision system used in this paper has a much slower frame rate, which introduces additional challenges in autonomous navigational task. In addition, the goal of this work is to investigate a fusion technique that would utilize the *pose* estimation of the vision system to correct the inherent error growth in the IMU system in a GPS deprived environment. Therefore, this system will act as an alternative navigation system until the GPS signal reception is recovered. It is obvious from this objective that this system must incorporate the absolute position in the

fusion algorithm rather than the relative position of the two sensor systems. However, estimating the absolute position from camera *pose* is impossible unless the absolute position of the initial camera position is known. Also, in achieving this one has to execute more complex IMU navigation algorithm and error modeling. The above developments differentiate the work presented in this work from the previously published work. In addition, here the authors introduce an outdoor calibration methodology which can be used in estimating the unique transformation that exists between the inertial sensor and the vision sensor. This is very important in Intelligent Transportation System (ITS) applications, i.e. the calibration must be performed outdoors, since the widely varying and more rigorous geo-positioning demands of outdoor conditions. These include fluctuating ambient lighting and temperature conditions, uncontrollable vibrations due to vehicle dynamics and roadway roughness and humidity changes etc. Furthermore, to address the problem of uncontrolled selection correspondences, a special validation criterion is employed to establish correct correspondences using epipolar geometry and correspondence motion fields. Also in this dissertation, the authors successfully compare a test run performed on an actual roadway setting to validate the presented fusion algorithm. In addition, to minimize the noise from the vision system, a pre-filter is used (Grewal 2001) to process the vision estimation prior to fusion with IMU data. The final results obtained from the fusion are successfully compared with positions obtained from a DGPS/INS integrated system installed in the same survey vehicle. Similarly, the rotations are confirmed with actual survey measurements.

1.3 Multi-Purpose Survey Vehicle

The sensor data for the experiments performed in this research was collected using a survey vehicle owned by the Florida Department of Transportation (FDOT) (Fig. 1) that is equipped with a cluster of sensors. Some of the sensors included in this vehicle are,

1) Navigational grade IMU
2) Two DVC1030C monocular vision sensors
3) A Basler L-103 camera
4) Laser profile system
5) Two Global Positioning System (GPS) receivers
6) A Distant Measuring Unit (DMI)
7) Pavement illumination system
8) Navigational computer

The original installation of sensors in this vehicle allows almost no freedom for adjustment of the sensors, which underscores the need for an initial calibration.

Figure 1: FDOT-multi purpose survey vehicle.

1.3.1 Inertial Measuring Unit (IMU)

The navigational grade IMU installed in the above vehicle (Fig. 1) was assembled by Applanix Corp., Toronto, Canada and installed by International Cybernetics Cooperation (ICC) of Largo, FL. It contains three solid state fiber-optic gyroscopes and three solid state silicon accelerometers that measure instantaneous accelerations and rates of rotation in three perpendicular directions. The IMU data is logged at any frequency in the range of 1Hz-200Hz. Due to its high frequency and the high accuracy, especially in short time intervals, IMU acts as the base for acquiring navigational data in data collection. Moreover, the IMU is a passive sensor which does not depend on any outside signal in measuring the accelerations and angular rates of the vehicle which enhances its role as the base data gathering sensor in the vehicle. However, due to the accelerometer biases and gyroscope drifts, which are unavoidable, the IMU measurements diverge. This is especially the case with the accelerometer on the vertical axis. Therefore, in order for the IMU to produce reliable navigational solutions its errors has to be corrected frequently. The position and velocities obtained from the GPS is generally used for this purpose.

Although the idea of fusion is quite intuitive the actual fusion process is intense and has to be performed at the system level since both the GPS and IMU systems have different errors which need to be minimized before integrating the two systems. FDOT multi purpose survey vehicle (MPSV) is also equipped with an INS/DGPS integrated system designed to overcome the afore-mentioned issue of error growth and measure the vehicle's spatial location and the orientation at any given instant. The integration of the INS and the DGPS system is achieved using a Kalman filtering based statistical

optimization algorithm which minimizes the long term error involved in the position prediction (Scherzinger, 2000). It also includes a computer that facilitates the navigation solution through its interaction with the DGPS unit. The block configuration of the GPS/IMU system is shown in Fig. 2.

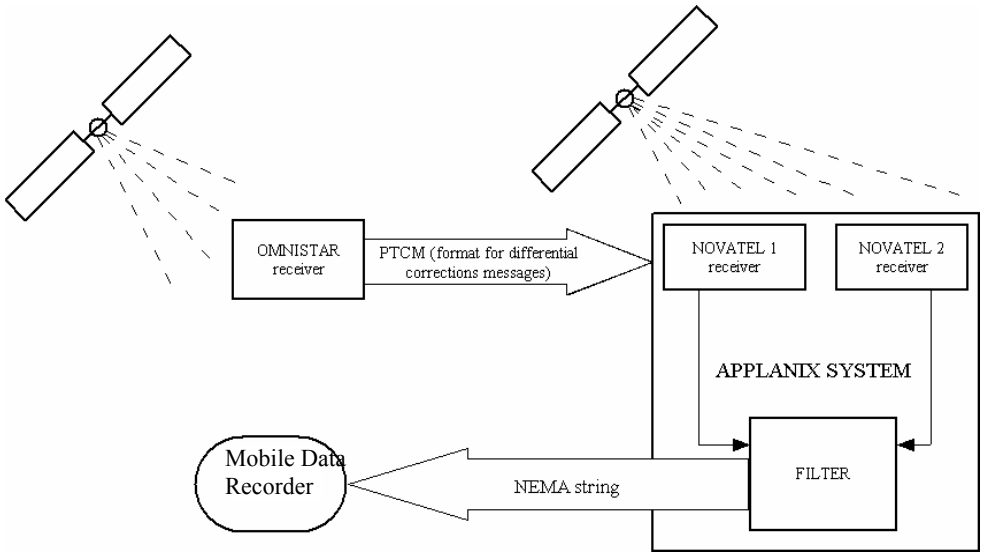


Figure 2: Block diagram of the GPS/INS system in the MPSV.

1.3.2 Lever Arm Effect on MPSV

The MPSV contains a multiple sensor system with each of the sensor systems having its coordinate origin at different positions. Out of these different sensor units some selected sensors are used in the aided navigation system. They are GPS, IMU and the camera system. When two or more systems with different coordinate origins are used in aided navigation architecture the translations measured by each sensor has to be brought to a one uniform origin in order to achieve accurate fusion of the sensors. Therefore, the system must be aware of the difference between the two sensor units in vector format for proper combination of the measured translations. This difference vector between two

sensor units is called the Lever arm and the corrections that have to be performed for the accelerations, velocities and distances due to this difference is called the lever arm effect.

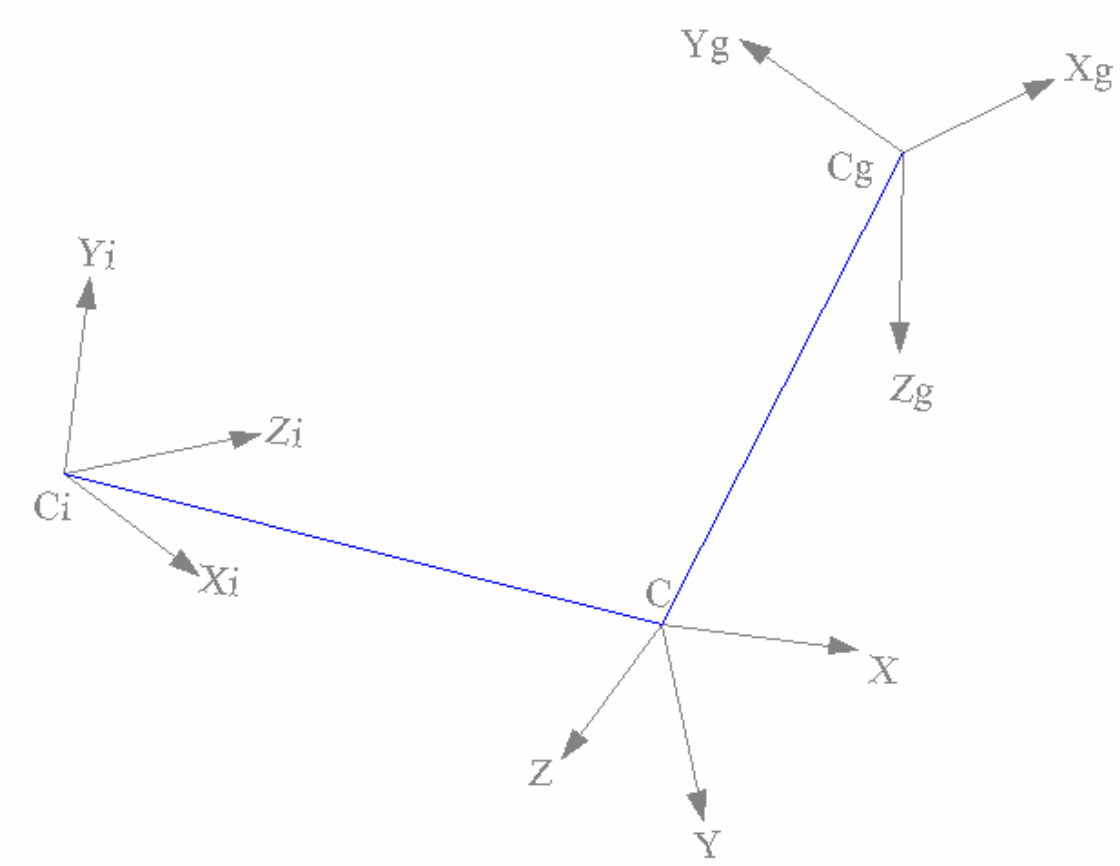


Figure 3: Illustration of lever arms of sensors.

In Fig. 3, C is the origin of the base coordinate frame while i, g represent two separate sensor units. Vectors CC_i and CC_g are the lever arms between the two sensor units with the base coordinate system.

1.3.3 Forward View and Side View Cameras

The FDOT survey vehicle also uses two high resolution (1300 x 1030) digital area-scan cameras for front-view and side-view imaging (Fig. 4) at a rate up to 11 frames per second. This enables capturing of digital images up to an operating speed of 60 mph.

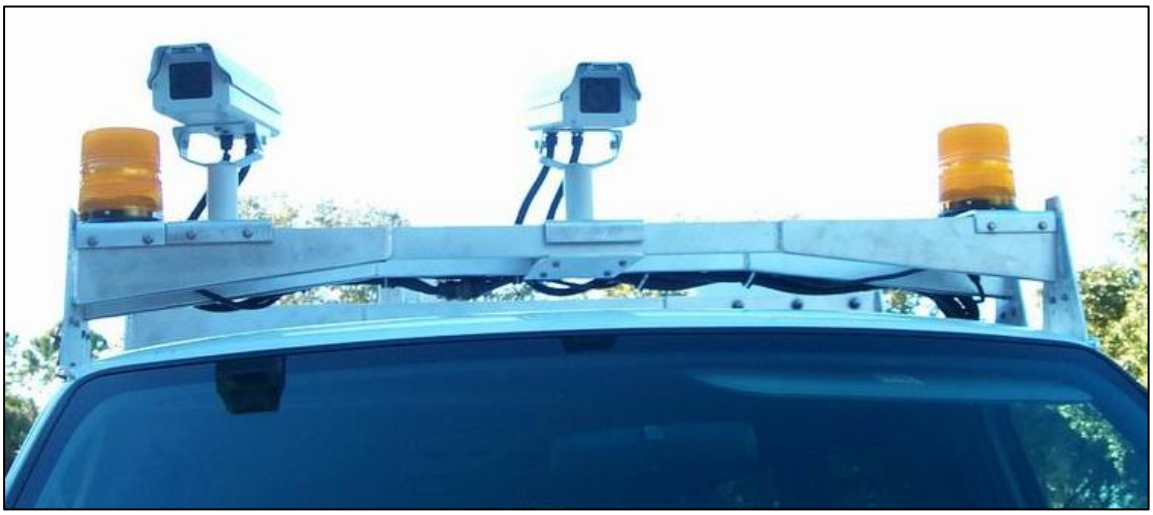


Figure 4: Forward and side view cameras.

The front-view camera with a 16.5 mm nominal focal length lens captures the panoramic view which includes pavement markings, number of lanes, roadway signing, work zones, traffic control and monitoring devices and other structures, Fig. 5.

Figure 5: An image from forward view camera.

Meanwhile, the side-view camera which uses a 25 mm nominal focal length lens is used to record right-of-way signs, street signs, bridge identification and miscellaneous safety features, Fig. 6.

Figure 6: An image from side view camera.

Both the forward view and side view cameras also utilize a 2/3 inch progressive scan interline Charge Coupled Device (CCD) sensor with a Bayer filter to minimize the

resolution loss. The CCD is an integrated circuit that contains many layers of semiconductor material, mostly Silicon, etched to perform different tasks such as capturing the light coming from the lens, storing these and various connections between each layer and out of the CCD chip. Out of these the light capturing layer consists of light sensitive capacitors coupled together as an array. These arrayed capacitors are also called picture elements (with 1339000 of them in forward and side view camera), which is abbreviated as pixels. Pixels define the smallest element that is used to make the image and the more pixels one has in an image the more detailed the image is. Due to this architecture of the CCD chip it captures more than 70% of the incident light on the chip making it far more efficient is photographic films which captures only less than 5% incident light. Additionally, in these cameras Bayer filter is used to filter the color of the incident light. This filter is closer to human eye in the respect that the filter has two green sensitive pixels for four pixels while the other two are red and blue.

1.3.4 Laser Profile System

One of the important sensors in measuring the cross-slope and grade of a roadway and rutting of the pavement is the laser profiler system installed in the FDOT Multi-Purpose Survey Vehicle (MPSV). The lasers in the MPSV project a laser beam onto the pavement and measures the distance between the pavement and the laser using the reflected beam. There are five lasers strategically placed on the MPSV in order to measure the roadway profile parameters mentioned earlier. Out of these, four lasers are on the front bumper, Fig. 7, of the vehicle allowing them to measure the height of the bumper from the pavement and the other one is at the rear bumper, Fig. 8, aligned with the middle laser

enabling grade measurement of the roadway. Of the four bumper lasers, one is at the right corner of the bumper while the others are located at distances of 33.25$^{//}$, 54.25$^{//}$ and 68$^{//}$ respectively from the right corner of the bumper.

Figure 7: Laser installation.

1.3.5 Pavement Imaging System

The pavement imaging system consists of a Basler L-103 line scan camera and pavement lighting system. The camera has a focal length of 15mm and a resolution of 2048x2942. The camera is mounted 9.25 ft above the pavement surface to capture an image of 14x20 ft respectively in width and length. In addition, the camera has two preset exposure values 1/19,000 and 1/40,000. The pavement lighting system is used to ensure that the pavement images captured are in excellent quality without and shadows. This system consists of 10, 150 W lamps.

Figure 8: Pavement lighting system and camera.

1.4 Motivation

Figs. 9 and 10 show two different runs performed by the Florida Department of Transportation (FDOT) Multi-Purpose Survey Vehicle (MPSV). MPSV is equipped with INS/DGPS system to measure the roadway characteristics using both the inertial system as well as the differential GPS systems. Two runs were made at different times on the same day on Interstate-4 in Volusia County. The first run was made with the GPS signal on, while the GPS signal was not available throughout the entire second run. Even though the manufacturer claims that the system would be stable even without the GPS signal, the obtained data refutes that. As seen from Fig. 9, when the GPS signal was available, the data obtained by the system correctly traces I-4 in Volusia County. However, Fig. 10 shows that navigation is meaningless when the GPS signal is not available the GPS location shows drastic errors. Therefore, finding an alternative error correction method for INS is essential.

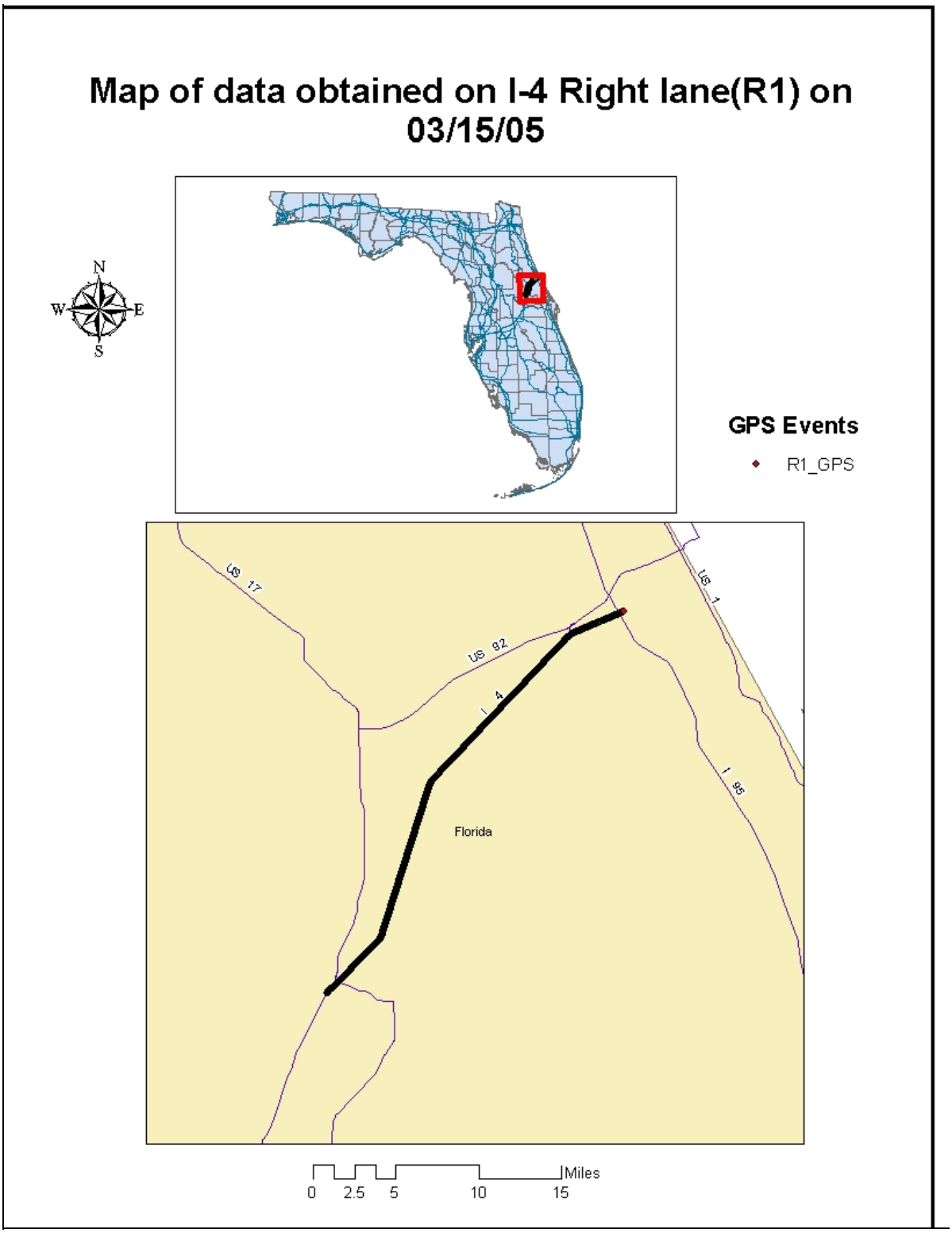


Figure 9: Test section drawn at I-4 when the GPS signal is available.

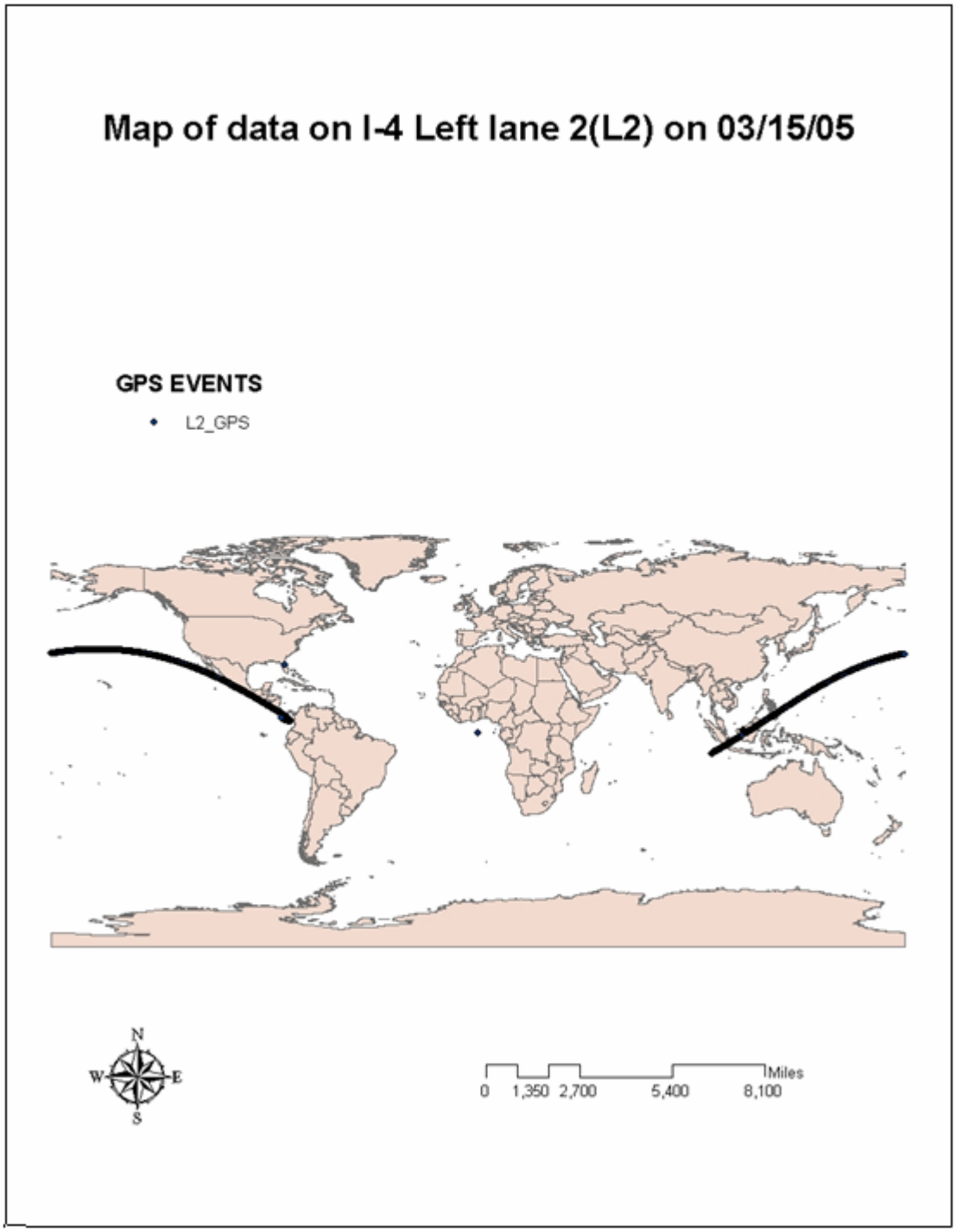


Figure 10: Test section drawn at I-4 when the GPS signal is not available.

Therefore, the specific objectives of this study are as follows:

1) Perform the intrinsic calibration of the vision system of the MPSV and estimate the unique transformation that exists between the vision and the Inertial Measurement Unit (IMU) installed in the vehicle.
2) Design a pre-filter for the noisy vision measurements so that it could be used in the fusion process.
3) Fuse vision and IMU sensor systems using a decentralized Kalman filter architecture and evaluate the results against the current IMU/GPS system and the ground truth.

1.5 Organization of Dissertation

This dissertation is organized as follows. The basic concepts of different coordinate frames used in the inertial navigation systems and camera systems are introduced in Chapter 2. Also the transformations involved with different coordinate frames are also included at the end of Chapter 2. This is followed by the inertial navigation fundamentals starting from formulating the navigational equations to inertial error propagations. Chapter 3 is devoted to the criteria developed by USF researchers in estimating and verifying geometric data obtained from the inertial readings on an actual roadway setting. A detailed description of fundamental computer vision theories used in this work is provided in Chapter 4. The fusion algorithm used in this work, i.e. Kalman filter, is introduced in Chapter 5 with detailed explanations of two particular filters developed in this research. The necessary mathematical formulations that form a critical part of the fusion and calibration algorithms are given in Chapter 6. Chapter 7 is devoted to

describing the experiments performed during the data collection process and all the results obtained during the experimentation. Finally, the conclusions related to the calibration and fusion algorithms are given in Chapter 8 followed by the appendices, which elaborate the relevant theoretical formulations.

CHAPTER 2 – FUNDAMENTALS OF INERTIAL NAVIGATION

2.1 Coordinate Frames Used in This Research

In Dead Reckoning (DR) navigation, which is the estimation of the present global position of an object by using the previous position, velocity, time and distance traveled by the vehicle, defining a coordinate system is very important. The obvious reason for this is the need for a proper reference system to describe a location on the Earth. In this chapter all the coordinate frames used in this research work are elaborated. For completeness the inertial frame, which is the fundamental coordinate frame, is also explained.

2.1.1 Inertial Frame (i-frame)

Inertial frame can be defined as a right handed coordinate frame where the Newton's laws of motion hold. Therefore, in an inertial frame, an object at rest will remain at rest and an object in motion will remain in motion with no acceleration in the absence of external forces. The object starts accelerating only when an external physical force is applied to the object. The dynamics of the motion of an object that is in an inertial frame can be formulated using Newton's laws of motion. In contrast, in non-inertial reference frames objects experience fictitious forces, which are the forces that act on the object due to the angular or linear accelerations of the reference frame, due to the acceleration of the reference frame but not because of any external force directly acting on the object.

Examples of two fictitious forces are centrifugal force and Coriolis force. Since these fictitious forces act on the object in non-inertial reference frames the dynamics of the motion cannot be modeled using Newton's laws of motion.

Defining a global inertial frame is an abstraction since any reference frame defined in the vicinity of planets, sun, moon and the other planets, are subjected to gravitational attractions, in varying magnitude according to the spatial location, making the reference frame a non-inertial frame. Because of this fact the classical definition of the inertial reference frame is given such that the origin of the reference frame is at the center of the Earth with 3rd axis of the frame parallel to the polar axis of the Earth, 1st axis realized by the distant *quasars,* which are extremely distant celestial objects that are of constant relative orientation, and the 2nd axis making a right handed coordinate system [Jekeli, 2000].

In addition to that there are various other definitions of the inertial frame obtained by modifying the Newton's laws of motion which are used in specific applications. In this research the inertial frame is taken as defined above and will be denoted as the i-frame hereafter.

2.1.2 Earth Centered Earth Fixed Frame (e-frame)

Earth Centered Earth Fixed (ECEF) frame is a right handed coordinate frame that is centered at the Earth's center of mass and with a fixed set of axes with respect to the Earth. The coordinate axes of the ECEF frame is defined as, the 3rd axis being parallel to

the mean and fixed polar axis, the 1st axis being the axis connecting center of mass of the Earth with the intersection of prime meridian (zero longitude) with the equator and the 2nd axis making the system a right handed coordinate frame. This is illustrated in Fig. 11. It is clear from the definitions of the i-frame and the e-frame that the difference between the two reference frames lie on the fact that e-frame is rotating with a constant angular velocity about its 3rd axis, which will be denoted as ω_e .

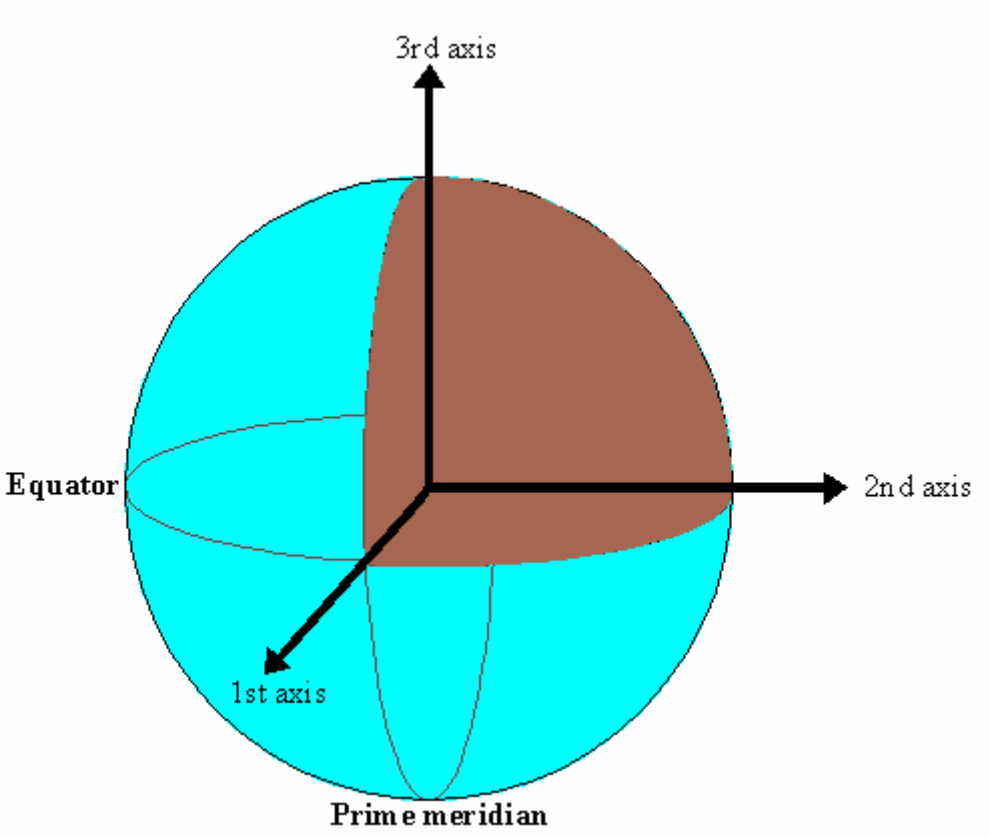


Figure 11: Illustration of ECEF coordinate frame.

2.1.3 Navigation Frame (n-frame)

The navigation coordinate frame is a locally leveled right handed reference frame, i.e. it changes from place to place. In other words this is a reference frame that moves with the object. This is the most commonly used frame in land navigation systems in providing the velocities along the directions of its coordinate axes. Even though the velocities are provided in the directions of the coordinate axes of the navigation frame this is not used to coordinatize the position of the vehicle since the reference frame moves with the object. Therefore, the primary use of the navigation frame is to provide local directions to the object.

The most commonly used navigation frame, or n-frame, in land navigation and geodetic applications is the reference frame defined with 3rd axis aligned with the normal to the Earth's surface where the object is at and towards the 'down' direction, i.e. same direction as gravity. The 1st axis points in the North direction, which is the direction parallel to the tangent to the meridian where the object is placed, and the 2nd axis points towards East making a right handed frame. This frame is also called as North-East-Down reference frame (NED frame). There are various other locally defined navigation frames available according to the application. However, throughout this dissertation the NED coordinate frame is taken as the navigation frame of the object.

2.1.4 Wander Angle Frame (w-frame)

There are instances which makes the NED frame not suitable for navigation due to singularities that occur, if the navigation solution is given in the NED frame. This happens at the poles where longitudes converge rapidly making the longitude rate, which is used to get the velocity along the 2nd axis of the reference frame, to become infinite. This problem can be overcome if the platform is rotated by an angle different from the longitude rate about the 3rd axis. This reference frame is called the wander angle frame, or the w-frame. Since the w-frame is only different from the n-frame by this rotation introduced about the 3rd axis, transformation between the two frames can be achieved simply by the rotation matrix given as,

$$C_n^w = \begin{pmatrix} \cos(\alpha) & \sin(\alpha) & 0 \\ -\sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{2-1}$$

Where, C_n^w is the transformation matrix from the n-frame to w-frame and α is the wander angle.

2.1.5 Body Coordinate Frame (b-frame)

This is the reference frame that the IMU measurements are made in a navigation system. This frame has its origin at a predefined location on the sensor and has its 1st axis towards the front, or forward movement direction of the system, 3rd axis towards the gravity direction and the 2nd axis toward the right side of the navigation system making a right handed coordinate frame (Fig. 12a). In this dissertation the body coordinate frame will be called as the b-frame.

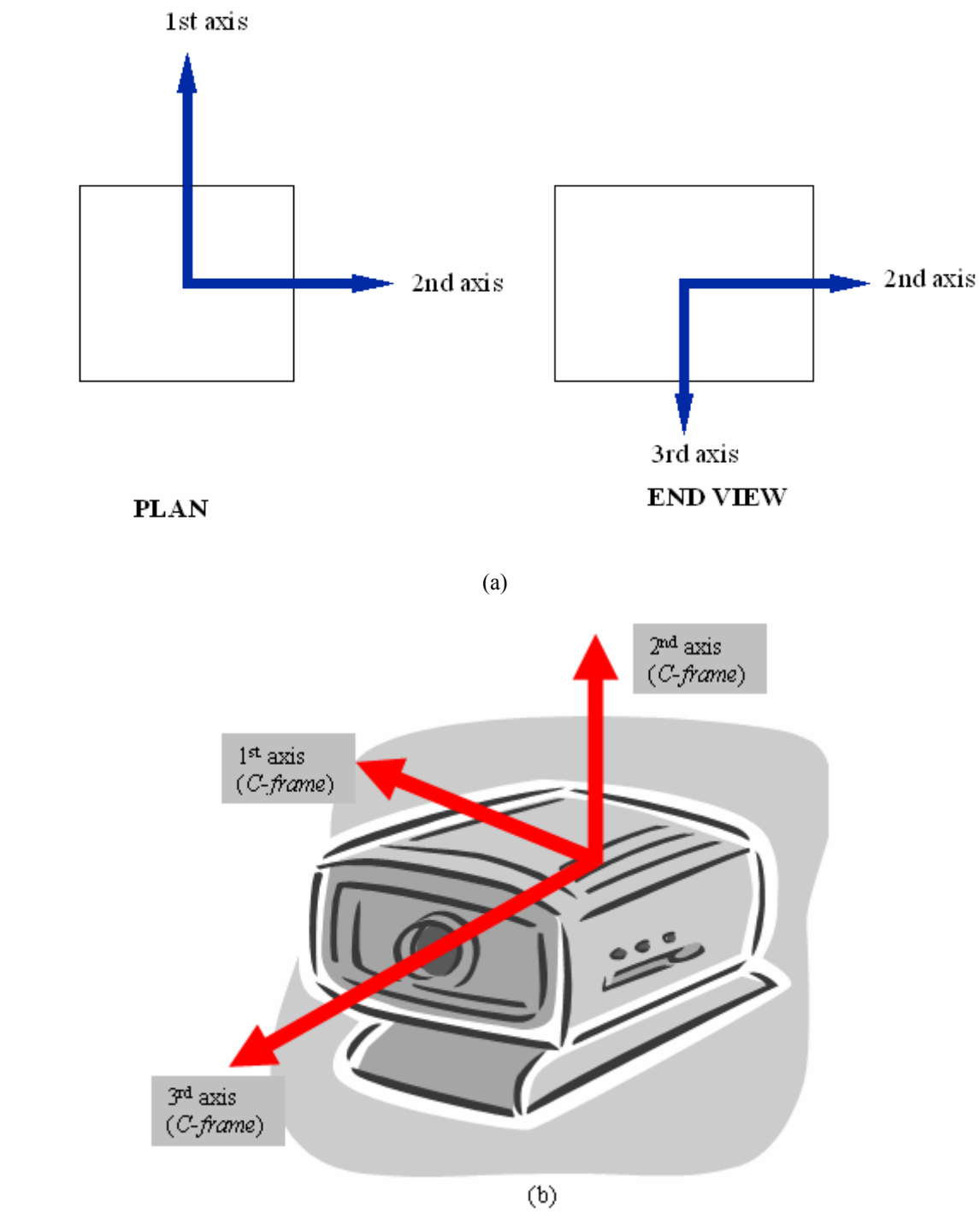


Figure 12: Illustration of local coordinate frames (a) body coordinate frame, (b) camera coordinate frame.

2.1.6 Camera Coordinate Frame (c-frame)

The reference frame that the vision measurements are made in is called the camera coordinate frame, or the c-frame. This reference frame for the camera system has its 3rd axis along the principal axis of the camera system, the 1st axis towards the right side of the image plane and the 2nd axis setup making the c-frame a right handed frame (Fig. 12b).

2.2 Representing Rotation

When considering the 3D navigation problem one of the most commonly encountered measurements are the translations of the object and the orientation, or rotation, of the object from the initial state to the next state. Out of these, translations can be represented by a simple arithmetic summation of differences of coordinates given in the vector form. But the orientation cannot be incorporated as easily as translations due to the mutual dependency of a rotation angle about one axis with respect to other axes in 3D space. But since the reference frames that are dealt here with are orthogonal and right handed reference frames, to describe relative orientation only three angles are needed and the transformation that exists between the two reference frames is also orthogonal. The transformation matrices defined for orthogonal, right handed frames have the property,

$$C_a^b (C_b^a)^T = I \Rightarrow (C_a^b)^{-1} = (C_b^a)^T \tag{2-2}$$

Where, *a*, *b* are two arbitrary frames and *C* denotes the transformation matrix between the two frames.

Therefore, it is clear that as long as the reference frames are orthogonal the transformation between the frames will be in SO(3) space and making the number of independent elements in C to be six, implying the relative orientation is only governed by three degrees of freedom.

There are various ways of representing the transformation between two reference frames. In this dissertation, two mostly utilized methods are described. Further information about transformations can be found on [Jekeli, 2000, Titterton, 1997].

2.2.1 Euler Angles

The most common method of representing the relative orientation between two frames is by a sequence of rotations about three perpendicular axes, for instance the three Cartesian axes. This was first introduced by the pioneering work of Leonhard Euler in describing the rigid body rotation in 3D Euclidean space. Due to this, the angles about each axes of the coordinate system are called Euler angles. However, according the field of application these Euler angles are given special nomenclature. As an example, in Aerospace Engineering they are called roll, pitch and yaw. In land navigation, they are called bank, attitude and heading. In this dissertation the Euler angles are also called as roll (Euler angle about x-axis), pitch (Euler angle about y-axis) and yaw (Euler angle about z-axis). This is illustrated in Fig. 13.

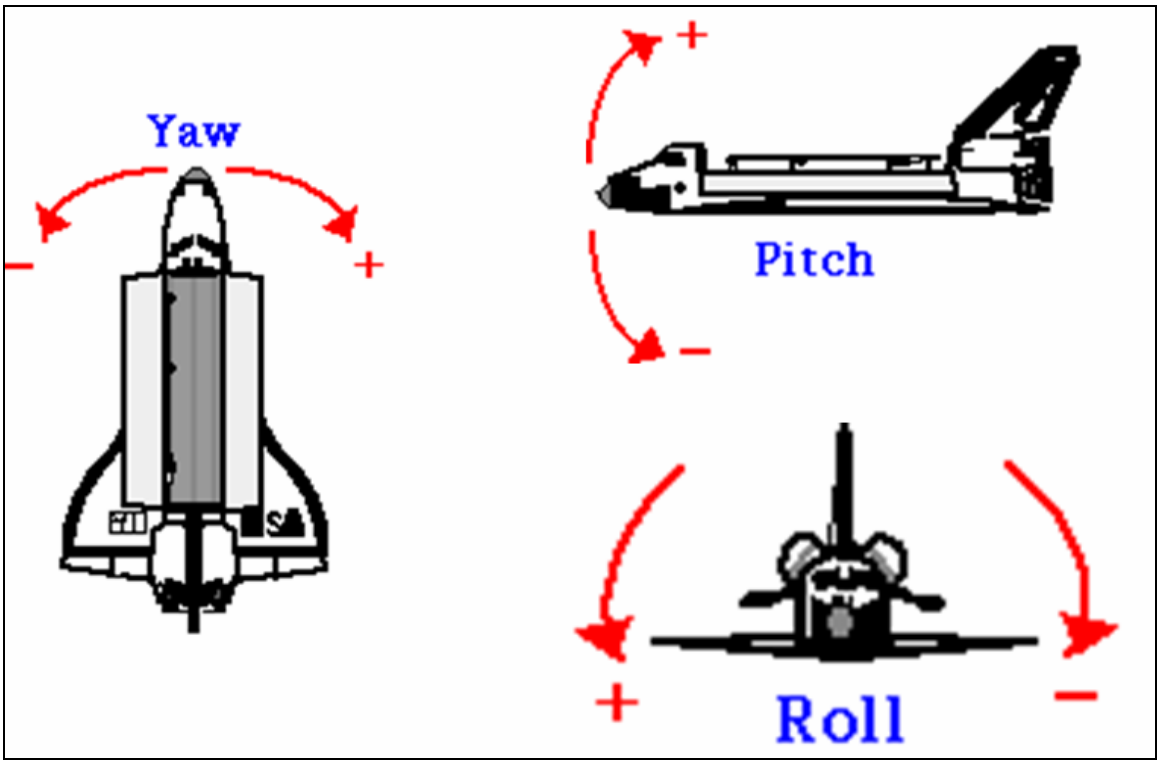


Figure 13: Illustration of roll, pitch and yaw.

These rotations can be written as rotation matrices along each of the axes and can be represented as;

$$R_x = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\theta) & \sin(\theta) \\ 0 & -\sin(\theta) & \cos(\theta) \end{bmatrix} \tag{2-3a}$$

$$R_y = \begin{bmatrix} \cos(\phi) & 0 & \sin(\phi) \\ 0 & 1 & 0 \\ -\sin(\phi) & 0 & \cos(\phi) \end{bmatrix} \tag{2-3b}$$

$$R_z = \begin{bmatrix} \cos(\psi) & \sin(\psi) & 0 \\ -\sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2-3c}$$

Where, θ, ϕ and ψ are respectively roll, pitch and yaw given in Euler angles and R_x, R_y and R_z are the rotation matrices given respectively about x, y and z axes. Using, Eqns. (2-3a -c) one can obtain one rotation matrix (R) as;

$$R = R_x * R_y * R_z \tag{2-4}$$

The Euler angles are on the SO(3), also known as Special Orthogonal group, which is a mathematical structure for rotations in 3D space representing the orthonormal matrices that preserve both the length and angles between vectors, or right handedness, once they were rotated while keeping the determinant as +1. But as can be seen from Eqn (2-4) the order at which the system rotates defines the final rotation of the system which makes the Euler angle rotations a complicated method of representing rotation in some situations. Additionally, when used in gyroscopes to measure the rotation the Euler angle rotations cause the Gimbal lock phenomenon, where one of the rotation references becomes inactive which reduces the 3D rotation into a 2D rotation problem.

2.2.2 Tate-Bryant Rotation Sequence

In order to avoid the confusion with different orders of rotation and define the Euler rotations in one constant sequence of rotation, the INS in MPSV [Pospac manual, POS/SV manual] uses a fixed rotation sequence called the Tate-Bryant sequence. The three rotations are once about x-axis, once about y-axis and once about z-axis. The Tate-Bryant rotation sequence is given as;

$$R_{T\text{-}B} = R_z R_y R_x \tag{2-5}$$

The Applanix system [Pospac manual, POS/SV manual] uses this rotation to define the Euler angle rotations.

2.2.3 Quaternions

A quaternion can be used to represent the rotation of an object using four parameters with three of them associated with a regular position vector and the other representing the angle of rotation as follows:

$$q = q_w + q_x i + q_y j + q_z k \tag{2-6}$$

Where, q_w defines the angle of rotation, q_x, q_y and q_z are the projections of the position vector along i, j, k directions respectively. Due to the extra dimensionality of the quaternion, four elements compared to three elements in Euler angles, the expression for the orientation becomes much simpler and easy to handle when comparing with Euler angles.

One must note that not all the quaternions can be used to denote an orientation. Only the unit quaternion, i.e. $1 = (q_w)^2 + (q_x)^2 + (q_y)^2 + (q_z)^2$, can be used to represent a rotation [Jekeli, 2000]. Since throughout this work both Euler angles and quaternions are used alternatively, depending on the need, it is important that one can convert Euler angles to quaternions and vise-versa. The conventional orientation matrix between the two reference frames can be derived in the quaternion space, for an arbitrary two spaces *a* and *b*, as follows (Titterton and Weston, 1997).

$$C_a^b = \begin{bmatrix} (q_w^2 + q_x^2 - q_y^2 - q_z^2) & 2(q_x q_y - q_z q_w) & 2(q_x q_z - q_y q_w) \\ 2(q_x q_y + q_z q_w) & (q_w^2 + q_y^2 - q_x^2 - q_z^2) & 2(q_y q_z - q_x q_w) \\ 2(q_x q_z - q_y q_w) & 2(q_y q_z + q_x q_w) & (q_w^2 + q_z^2 - q_x^2 - q_y^2) \end{bmatrix} \quad (2\text{-}7)$$

Formulation of the quaternion-based orientation matrix (2-7) using known Euler angles and the inverse operation of obtaining the respective Euler angles using a known quaternion orientation matrix can both be performed conveniently (Titterton and Weston, 1997; Shoemake, 1985). For instance, if the orientation between two reference frames are given in Euler angles they can be converted to quaternions using Eqns (2-4) and (2-8). Considering the order of rotation of Euler angles R_z R_y R_x the quaternions can be given as,

$$\begin{aligned} q_w &= \cos(\theta/2)\cos(\phi/2)\cos(\psi/2) + \sin(\theta/2)\sin(\phi/2)\sin(\psi/2) \\ q_x &= \sin(\theta/2)\cos(\phi/2)\cos(\psi/2) - \cos(\theta/2)\sin(\phi/2)\sin(\psi/2) \\ q_w &= \cos(\theta/2)\sin(\phi/2)\cos(\psi/2) + \sin(\theta/2)\cos(\phi/2)\sin(\psi/2) \\ q_w &= \cos(\theta/2)\cos(\phi/2)\sin(\psi/2) - \sin(\theta/2)\sin(\phi/2)\cos(\psi/2) \end{aligned} \quad (2\text{-}8)$$

Where, θ, ϕ and ψ are respectively the roll, pitch and yaw given in Euler angles. Similarly, it is possible to convert a given quaternion to Euler angles. The procedure followed here is to first estimate the transformation matrix given by Eqn (2-4) and then determine the Euler angles.

In this work the quaternion representation is used extensively to various applications such as interpolating between two rotations and estimating the orientation from a sequence of images (Taylor and Kriegman., 1995) etc. The main reasons for this choice are the complexities introduced by the Euler angles in interpolation and the ability of the

quaternion method to eliminate the Gimbal-lock a phenomenon which arises from the misalignment of gimbals.

2.3 Coordinate Transformations

When designing a navigation system, dealing with different reference frames is unavoidable. This is due to the fact that the measurements are obtained in one reference frame, b-frame, while the navigation solution must be given in another reference frame, n-frame or e-frame. Therefore, it is essential that one be able to transform measurements or data in one reference frame to any other reference frame as required. In this section transformation between the important reference frames are considered.

2.3.1 Transformation from i-frame to e-frame

Since the origins of the i-frame and the e-frame are at the Earth's center of mass this particular transformation is relatively simple. Furthermore, the difference between the above two reference frames is a simple rotation about the 3rd axis, to compensate for the Earth's rotation. This can be shown as,

$$C_i^e = \begin{pmatrix} \cos(\omega_e t) & \sin(\omega_e t) & 0 \\ -\sin(\omega_e t) & \cos(\omega_e t) & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{2-9}$$

Where, ω_e denotes the rate rotation of the Earth. It is clear that the angular velocity vector between the two reference frames can be given as,

$$\omega_{ie}^{e} = \begin{pmatrix} 0 & 0 & \omega_e \end{pmatrix}^T \tag{2-10}$$

In Eqn (2-10) ω_{ie}^{e} is used to denote the angular velocity vector of the e-frame with respect to i-frame, coordinatized in the e-frame.

2.3.2 Transformation from n-frame to e-frame

The e-frame and the n-frame have two different origins, i.e. they are not concentric. Due to this reason the transformation between the above two frames are relatively more complex than the transformation between i-frame and e-frame. Since both reference frames are right handed and the selected n-frame is given in NED directions, this transformation can be derived using Euler angles, first by rotating the coordinate system about the 2nd axis by an angle of $\left(\frac{\pi}{2} + \eta\right)$ and then rotating it about the 3rd axis by $(-\lambda)$.

The overall transformation matrix can be given as,

$$C_n^e = \begin{pmatrix} -\sin(\eta)\cos(\lambda) & -\sin(\lambda) & -\cos(\eta)\cos(\lambda) \\ -\sin(\eta)\sin(\lambda) & \cos(\lambda) & -\cos(\eta)\sin(\lambda) \\ \cos(\eta) & 0 & -\sin(\eta) \end{pmatrix} \tag{2-11}$$

Where, η and λ represent the geodetic latitude and the longitude respectively. Also the angular rates vector is given [Jekeli, 2000] as,

$$\omega_{en}^{n} = \begin{pmatrix} \dot{\lambda}\cos(\eta) & \dot{\eta} & \dot{\lambda}\sin(\eta) \end{pmatrix}^T \tag{2-12}$$

Where, $\dot{\lambda}$ and $\dot{\eta}$ denote the first time derivative of the latitude and the longitude. Also one can find the angular rotation rates of the n-frame with respect to the i-frame using the simple vector manipulations and known vectors as,

$$\begin{aligned} \omega_{in}^{n} &= \omega_{ie}^{n} + \omega_{en}^{n} \\ \omega_{in}^{n} &= ((\dot{\lambda} + \omega_e)\cos(\eta) \quad -\dot{\eta} \quad -(\dot{\lambda} + \omega_e)\sin(\eta))^T \end{aligned} \tag{2-13}$$

2.3.3 Transformation from b-frame to n-frame

Transformation between the b-frame and the n-frame can also be obtained as a sequence of rotations in Euler angles. In [Titterton, 1997, Jekeli, 2000] this transformation is given as,

$$C_b^n = R_3(-\psi)R_3(-\phi)R_3(-\theta) \tag{2-14}$$

Where θ, ϕ and ψ are respectively roll, pitch and heading of the vehicle measured by the IMU. Although it is relatively easy to obtain the transformation between the b-frame and the n-frame, in practice it must be noted that the result will depend on the mechanization of the configuration. This can be given in matrix format as,

$$C_b^n = \begin{bmatrix} c(\psi)c(\phi) & c(\psi)s(\phi)s(\theta) - c(\theta)s(\psi) & c(\psi)s(\phi)c(\theta) + s(\psi)s(\theta) \\ s(\psi)c(\phi) & c(\theta)c(\psi) + s(\theta)s(\phi)s(\psi) & c(\theta)s(\phi)s(\psi) - s(\theta)c(\psi) \\ -s(\phi) & c(\phi)s(\theta) & c(\theta)c(\phi) \end{bmatrix} \tag{2-15}$$

In Eqn (2-15) *c(),s()* represents cosine and sine respectively.

2.4 Introduction to Inertial Measuring Unit

Navigation can be defined as the estimation of the current position of an object with respect to a pre-specified final destination. There are many forms of navigation available [Grewal 2001]. Of these, one of the most commonly used methods is called dead reckoning which corresponds to knowing the initial position and orientation (pose) of the vehicle and obtaining the current position by estimating the heading information and the acceleration information. One of the key instruments that is used in dead reckoning is the Inertial Measurement unit (IMU) which uses gyroscopes and accelerometers. Gyroscopes are used to measure the rotation while the accelerometers are used to measure the acceleration of the object thereby giving the velocity and the displacement. Typically, an IMU contains three accelerometers and three gyroscopes fixed in three perpendicular directions allowing the measurements to be made in six degrees of freedom. The first use of gyroscopes for navigation purposes started in 1911 when it was used in iron ships for automatic steering and since then they have evolved rapidly. The first IMU was used as a navigational sensor in the 1950's. Since then inertial measurement units have been used in almost all the navigation systems due to their high sensitivity and accuracy under certain conditions. There are two common types of IMU used in navigation systems;

1) Strapdown IMU (Fig. 14 a)
2) Gimbaled IMU (Fig. 14 b)

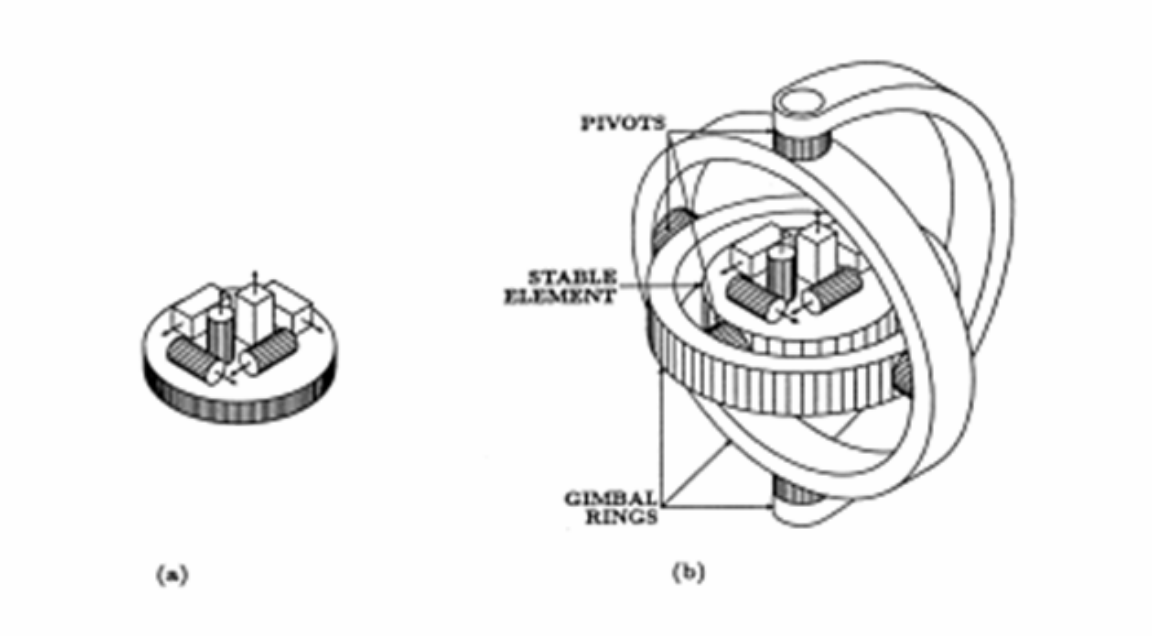


Figure 14: (a) Strapdown IMU and (b) Gimbaled IMU (Grewal 2001).

2.4.1 Gimbaled IMU

As shown in Fig 14 (b) the gimbaled system consists of three gimbals that are pivoted to each other so as to make the input axes for the three rings form three perpendicular directions that one wishes to measure the rotations. When the object the gimbaled IMU is attached to is subjected to rotations about any axis, the three gimbals isolate these rotations making an element placed in between the gimbals stable at all times. This isolation from the rotation is achieved not only by the gimbals but also the inertia of the stable element. Ideally these two alone can isolate the rotation but in practice the friction in the bearings could cause a drift.

Therefore, a servo motor is used in many cases to isolate the rotations of the body from the stable element. Since the gyroscopes measure the angular rate, i.e. the output is small angles, and this angle is measured in voltage using an electric pickoff device. This

voltage will generate a current that will used to drive a servo motor which will bring the stable element back to the same frame isolating the rotations. Due to this arrangement of the stable element, if one places the accelerometers on the stable platform, between the gimbals, the accelerometers always oriented in the same coordinate frame that they were oriented at the start of the navigation process.

2.4.2 Strapdown IMU

As the name suggests, in the strapdown case, the IMU is rigidly fixed to the platform making the IMU go through all the translations and rotations the platform which it is fixed is going through during navigation. Since the accelerometers and gyroscopes are subjected to all the dynamics the vehicle is experiencing, their performances tend to degrade compared to the gimbaled counterpart. The typical errors that cause this degradation are errors in gyroscopes due to cross-coupling of angular rates and angular acceleration, errors in accelerometers due to lever arm effect arising from vehicle rotations and errors due to integration of gyroscope and accelerometer readings in a rotating frame. The frame that those readings are integrated must be non-rotating causing *coning* and *sculling* errors in gyroscopes and accelerometers respectively. Therefore, a navigational computer is crucial in the case of a strapdown IMU in order to correct for these errors and provide accurate angular rates and accelerations. Additionally, the navigational computer is used to transform the angular rates and accelerations that are measured in body coordinates to the navigation frame.

Since the strapdown IMU measurements are made in the body coordinate frame, the highest accuracy that could be achieved from early strapdown systems were worse than that of the gimbaled IMU. As the technology in developing gyroscopes such as ring laser gyros and fiber optic gyros and accelerometers such as silicon accelerometers advances, the strapdown IMU system measurements approached gimbaled IMU measurements. Moreover, due to its robustness with no moving parts, compactness, light weight, low power consumption, less maintenance and low cost makes the strapdown IMU a better choice in most navigational applications than gimbaled IMU. As the technology is developed the performance and the accuracy of the strapdown IMU also enhanced. There are four major grades of strapdown IMU available according to the intended usage. They are described in Table 2.

Table 2: Comparison of different grades of IMU (Shin 2005, Grewal 2001).

Grade		Navigation	Tactical	Automotive	Consumer
Position error		1.9 (km/hr)	19-38 (km/hr)	≈2 (km/min)	≈3 (km/min)
Gyro	bias (deg/hr)	0.005-0.01	1-10	180	360
	scale factor (ppm)	5-50	200-500		
	noise (deg/hr/$\sqrt{}$Hz)	0.002-0.005	0.2-0.5		
Accel	bias (μg)	5-10	200-500	1200	2400
	scale factor (ppm)	10-20	400-1000		
	noise (μg/hr/$\sqrt{}$Hz)	5-10	200-400		

2.4.3 Fiber Optic Gyroscopes

Fiber optic gyroscopes (FOG) are based on kinematics rather than dynamics in measuring the angular rates. Instead of the spinning mass in mechanical gyroscopes, in FOG light acts as the sensor element. FOG is a closed path optical fiber combined with a coupler, a polarizer, a photo-detector and some digital signal processing equipment that are used in estimating the phase shift of two light beams after traveling in opposite directions through the loop (Fig. 15). When a broad spectrum of light is passed through the coupler it directs the light from one fiber to another optical fiber. Then the light beam enters the polarizer which passes only light of the proper polarization, which is useful to assure total reciprocal optical path lengths for counter-propagating light beams, to the optical fiber. This is then split into two waves that propagate in opposite directions. Once the two beams exit the loop after propagating they are combined and sent to the photo-detector. The photo-detector is used to convert the output light intensity into an electric signal which can be related to the phase difference of the two light beams. The phase difference can be related to the angular rate by;

$$\Delta\Phi = \frac{4A}{c}\omega \tag{2-16}$$

Where, Φ is the phase of the light beam, A is the area of the loop, c is the speed of light and ω is the angular rate of the loop. It is clear from the Eqn (2-16) that the rotational sensitivity of the FOG can be increased by increasing the length of the fiber optic cable. Eqn (2-16), also called the rotational sensitivity of the FOG, is due to a phenomenon

called *Sagnac effect* which not only FOG but all the optical gyroscopes are based on. Sagnac effect is the lengthening (or shortening) of the distance that a light beam has to propagate around a closed path to reach the emitter in a frame that rotates with respect to the inertial frame. Since the phase difference between the clockwise and counter-clockwise waves is measured, these types of FOG are also called interferometric type FOG (IFOG).

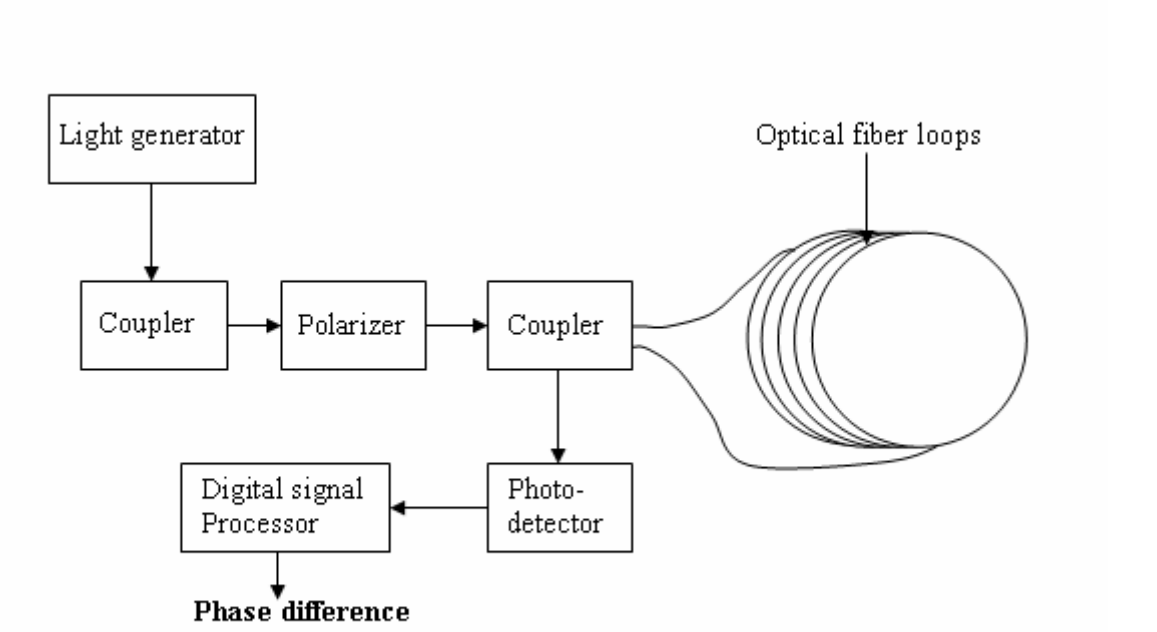


Figure 15: Schematics of IFOG.

By definition the FOG has a single degree of freedom and to measure the angular rates on three perpendicular directions the IMU needs three separate FOG s placed along the three axes.

2.4.4 Accelerometers

Accelerometer is an instrument that is used to measure the specific force and hence it is one of the prominent sensors is terrestrial navigation. The specific force can be defined as the acceleration caused by the real forces applied on a unit mass. The gravitational force exerts no real force on the system. Therefore, an accelerometer cannot sense the gravitational acceleration even though the system is accelerating. So an accelerometer can be defined as a device that senses the specific force on a known mass for the purpose of measuring acceleration. Since the main task of the accelerometer is to measure the specific force on the proof mass, different techniques can be used in measuring the specific force including the Micro Electro-Mechanical Systems (MEMS) accelerometers. The latter type is used in the FDOT MPSV. Construction of accelerometers is beyond the scope of this dissertation and the interested reader can see a more elaborate discussion of different types of accelerometers and the dynamics behind them in (Jekeli, 1999).

2.5 Estimating Navigation Equations

Inertial measuring unit in the MPSV is a strapdown IMU with three, Single Degree of Freedom (SDF) silicon, MEMS accelerometers and three fiber optic gyroscopes aligned in three mutually perpendicular axes. When the vehicle is in motion the accelerometers measure the specific force and the gyroscopes measure the rate of change of angle of the vehicle. Therefore, it is clear that in order to navigate the MPSV with a known initial position, i.e. to dead reckon, one has to integrate the outputs of the accelerometers and gyroscopes. This means that one has to solve a second order differential equation for the

accelerometers and a first order differential equation for gyroscopes both with respect to time in order to obtain the necessary position and attitude parameters.

Although the final navigation solution for the velocity is given in the n-frame, and for the position in the e-frame, the measurements are made in the b-frame. Hence, one needs to find the transformation between the n-frame and the b-frame at that particular instance. In other words, the measurements from the accelerometers and the gyroscopes are in body coordinate frame but the final position and orientation has to be given in ECEF. So the necessary transformation can be separated in to two steps,

1) Transformation between b-frame and n-frame.
2) Transformation between n-frame and ECEF frame.

Out of these the latter is easily obtained using Eqn (2-11), but estimating the transformation between the b-frame and n-frame is not straight-forward. Therefore, one of the important tasks is to find the transformation between the body frame and the navigation frame, which varies according to the vehicle movement. In order to obtain the transformation matrix between the b-frame and the n-frame one has to setup a differential equation in Euler angles which are measured by the gyroscopes. But due to the inherent problems of Euler angles, such as singularities at pitch angle approaching $\pm\frac{\pi}{2}$ and the complexity introduced to the problem due to the trigonometric functions, quaternions are preferred in deriving this differential equation. Therefore, in this work quaternions were used in deriving this transformation as illustrated below.

The time propagation equation in terms of quaternions can be given as;

$$\dot{q} = \frac{1}{2} A q \tag{2-17}$$

Where, q is defined as Eqn (2-6) and the skew-symmetric matrix A can be given as;

$$A = \begin{pmatrix} 0 & \omega_1 & \omega_2 & \omega_3 \\ -\omega_1 & 0 & \omega_3 & -\omega_2 \\ -\omega_2 & -\omega_3 & 0 & \omega_1 \\ -\omega_3 & \omega_2 & -\omega_1 & 0 \end{pmatrix} \tag{2-18}$$

Where, $\omega_{nb}^{b} = \begin{pmatrix} \omega_1 & \omega_2 & \omega_3 \end{pmatrix}^T$ is the angular rate of the b-frame with respect to the n-frame given in the b-frame and ω_j represents the angular rates about the j^{th} axis (j=1, 2, and 3). This ω_{nb}^{b} can be estimated by (vector) addition of the gyroscope readings Eqn (2-19) at each instance and the rate of change of the n-frame with respect to the i-frame. The angular rates measured by the gyroscopes in IMU are rates of change of angles of the b-frame with respect to i-frame, i.e. ω_{ib}^{b}. Hence transformation between the two frames can be given (Jekeli 1999) as,

$$\omega_{nb}^{b} = \omega_{ib}^{b} - C_{n}^{b} \omega_{in}^{n} \tag{2-19}$$

The measurements from the IMU gyroscopes are ω_{ib}^{b} and ω_{in}^{n} which are the angular rate of the n-frame and can be estimated by the transformation described in Eqn (2-13) if the positions are given in geodetic coordinates. Once ω_{nb}^{b} is found using Eqn (2-19) one can solve the differential equation given in Eqn (2-17) to estimate the quaternion. This

quaternion can be used to estimate the transformation matrix between the n-frame and b-frame from Eqn (2-7) in usual matrix notation.

The numerical integration used in estimating C_n^b must take into consideration the errors that could originate from integrating the angular rates on a rotating frame. This error is called the *sculling* error in gyroscopes. This error can be eliminated by reducing the time interval in the numerical integration process. Reducing the time interval adds more complexity to the computation process. Therefore, a suitable time interval has to be decided according to the level of accuracy needed for the particular application.

On the other hand, accelerometers in the IMU measure the specific force which can be given as,

$$\ddot{x}^i = g^i(x^i) + a^i \tag{2-20}$$

Where, a^i is the specific force measured by the accelerometers in the inertial frame and $g^i(x^i)$ is the acceleration due to the gravitational field which is a function of the position x^i. From Eqn (2-17) and Eqn (2-20) one can deduce the navigation equations of the vehicle in any frame by taking into consideration the different transformations between frames introduced in Section 2.4. In this work all the navigation solutions are given along the directions of the n-frame. In clarifying, what is desired in terrestrial navigation are the final position, velocity, and orientations provided in the local North-East-Down frame although the measurements are made in another local frame, the b-frame. This is not possible since the n-frame also moves with the vehicle making the vehicle stationary in

the horizontal direction on this local coordinate frame. Therefore, the desired coordinate frame is the fixed ECEF where the integration can be performed. To provide the output along the local n-frame the ECEF frame can be coordinatized along the NED, the n-frame, directions. Hereafter in this dissertation any parameter with the superscript *n* denotes the value of that parameter in the ECEF frame but coordinatized along the n-frame.

Since both the frames considered here are non-inertial frames, frames that are rotating and accelerating, one has to take into consideration the Coriolis force that affects the measurements. Coriolis acceleration is the change of acceleration due to rotation of the measuring coordinate frame with respect to the inertial frame. Additionally, since any object on the surface of Earth is subjected to gravitational forces both from the Earth itself and all other planets in the solar system one has to consider the gravitational force exerted on the vehicle when deriving the systems equations. Once the effects of these forces are considered the navigation equation can be given in the following form,

$$\frac{d}{dt}v^n = a^n - (\Omega_{in}^n + \Omega_{ie}^n)v^n + g^n \tag{2-21a}$$

$$\frac{d}{dt}x^n = v^n \tag{2-21b}$$

Where, Eqn (2-21a) is for the acceleration of the vehicle in terms of velocity and Eqn (2-21b) is for the velocity in terms of position. The second and third terms in Eqn (2-21a) are respectively the Coriolis acceleration and the gravitational acceleration on the vehicle.

The vector multiplication of angular rate is denoted as Ω. This represents vector multiplication of the measured angular rates and also called as an axial vector [Jekeli, 1999] having the form,

$$\Omega = [\omega \times] \tag{2-22a}$$

$$\Omega = [\omega \times] = \begin{pmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{pmatrix} \times = \begin{pmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{pmatrix} \tag{2-22b}$$

Although Eqns (2-21) describes the complete vehicle dynamics this navigation solution does not provide the orientation of the vehicle. It provides only the position of the vehicle. But for the automatic navigation of land vehicles orientation of the vehicle is also vital and important in estimating the true location of the vehicle. This can be given as (Jekeli 1999, Titterton 1997);

$$\frac{d}{dt} C_b^n = C_b^n \Omega_{nb}^n \tag{2-23}$$

In Eq. (2-23) Ω_{nb}^n can be obtained using Eqn (2-22). Therefore once the gyroscope and accelerometer measurements are obtained one can set up the complete set of navigation equations by using Eqns. (2-21) – (2-23). From the navigation equations given in Eqn (2-21) one can obtain the position of the vehicle and the traveling velocity by integrating the differential equations with respect to time. As in the case of the integration of gyroscopic readings the complexity of the integration algorithm used in obtaining the velocities and distances depend on the needed accuracy. In general, both the second and third terms of

Eqn (2-21a) are smaller and less variable relative to the acceleration term. This is in fact true for most of the terrestrial navigation tasks because of the relatively slow rate of rotation of the n-frame with respect to i-frame. Therefore, in obtaining the velocity one can use a higher order algorithm in only estimating the acceleration term while the Coriolis and gravitational terms can be integrated using a first order algorithm. This can be expressed as,

$$\Delta v^n = \int_{\Delta t} C_b^n a^b dt + \left((\Omega_{in}^n + \Omega_{ie}^n)v^n + g^n\right)\Delta t \tag{2-24}$$

Integration of the first term in Eqn (2-24) can be performed as;

$$\int_{\Delta t} C_b^n a^b dt = C_b^n \begin{pmatrix} 1 & 0.5\omega_3 & -0.5\omega_2 \\ -0.5\omega_3 & 1 & 0.5\omega_1 \\ 0.5\omega_2 & -0.5\omega_1 & 1 \end{pmatrix} \Delta a^b \tag{2-25}$$

Where, $\omega_{nb}^b = \begin{pmatrix} \omega_1 & \omega_2 & \omega_3 \end{pmatrix}^T$ have been estimated earlier. The gravitational acceleration can be estimated using the definition of the geoid given in WGS1984 definition [shin 2005]. Then the velocity of the vehicle at time step (k+1) can be given as,

$$v_{(k+1)}^n = v_{(k)}^n + \Delta v^n \tag{2-26}$$

Once the velocities are obtained from the Eqn (2-26) the positions can be deduced with ease. But one important factor that one has to take into consideration when obtaining the

positions is that it has to be in geodetic coordinates, i.e. in terms of latitude, longitude and height. It is essential to express the positions in geodetic coordinates not only because the positions provided in latitude, longitude and height would be more appropriate especially in terrestrial navigation but also in estimating the transformation between the ECEF frame and n-frame that was utilized in Section 2.4. The positions can be obtained by integrating the Eqn (2-26) and can be converted to the geodetic coordinate frame as,

$$\phi_{(k+1)} = \phi_{(k)} + \frac{(v_N^n)_k \Delta t}{(M_k + h_k)} \tag{2-27}$$

$$\lambda_{(k+1)} = \lambda_{(k)} + \frac{(v_E^n)_k \Delta t}{(N_k + h_k)\cos(\phi_k)} \tag{2-28}$$

$$h_{(k+1)} = h_{(k)} - (v_D)_k \Delta t \tag{2-29}$$

Where, v_N, v_E, v_D are respectively the velocities estimated in Eqn (2-26) coordinatized along the local North, East and Down directions while ϕ is the latitude, λ is the longitude and h is the height. And M and N are respectively the radius of curvature of the Earth at the meridian and the radius of curvature of the Earth at the prime vertical where the vehicle is located. They are given as follows,

$$N = \frac{p}{\sqrt{(1 - e^2 \sin^2 \phi)}} \tag{2-30}$$

$$M = \frac{p(1 - e^2)}{(1 - e^2 \sin^2 \phi)^{3/2}} \tag{2-31}$$

Where p is the semi-major axis of the Earth and e is the first eccentricity of the ellipsoid.

2.6 IMU Error Model

Since the navigation solution was derived from the measurements obtained from gyroscopes and accelerometers which are complex sensor units with different characteristics, errors caused by these sensors such as measurement, manufacturing and bias errors will be propagated to the navigation solution making the navigation solution erroneous. Therefore, it is important to develop proper dynamics and appropriate error models in modeling the system error characteristics to minimize the effects of these errors in the navigation solution. This error could have higher order terms but in this work only the first order error terms are considered implying that the higher order terms contribute only a small portion of the error compared to the first order terms. In addition, by selecting only the first order terms the error dynamics of the navigation solution becomes linear with respect the errors (Jekeli 1999, Titterton 1997).

Error dynamics used in this work were derived by differentially perturbing the navigation solution by a small differentials and then obtaining only the first order terms of the perturbed navigation solution. As in the derivation of the navigation solutions (Section 2.3) perturbation equations can be given in any coordinate frame. But here the perturbed equations are given in the e-frame and resolved in the n-frame. Therefore, by perturbing the Eqns (2-21) one can obtain the linear error dynamics for the IMU in the ECEF frame coordinatized along the n-frame. These error difference equations take the following form (Jekeli 1999, Titterton 1997),

$$\delta\ddot{x} = -\omega_{en}^{n} \times \delta x^{n} + \delta\varphi \times v^{n} + \delta v^{n} \tag{2-32}$$

Where, δ denotes the small perturbation given to the position differential equation, Eqn (2-21b) and φ denotes the rotation vector for the position error. And the vector multiplication denoted by x is same as given in Eqn (2-22b). Similarly, if one perturbs Eq.(2-21a) the following first order error dynamic equation can be obtained,

$$\delta v^n = C_b^n \delta a^b + C_b^n a^b \times \varepsilon + \delta g^n - (\omega_{ie}^n + \omega_{in}^n) \times \delta v^n - (\delta\omega_{ie}^n + \delta\omega_{in}^n) \times v^n \qquad (2\text{-}33)$$

Where, ε denotes the rotation vector for the error in the transformation between the n-frame and the b-frame. The first two terms on the right hand side of the Eqn.(2-33) are respectively due to the errors in specific force measurement and errors in transformation between the two frames, i.e. errors in gyroscope measurements. In other words when perturbing the navigation equations, the basic procedure is to add small differentials to quantities that are obtained or derived from the accelerometer and gyroscope measurements. When Eqn. (2-23) is perturbed the following equation is obtained,

$$\delta\dot{\Psi} = -\omega_{in}^n \times \varepsilon + \delta\omega_{in}^n - C_b^n \delta\omega_{ib}^b \qquad (2\text{-}34)$$

Eqns(2-32) – (2-34) are linear with respect to the error of the navigation equation. Therefore, they can be used in a linear Kalman filter to statistically optimize the error in propagation. This will be discussed further in Chapter 6.

CHAPTER 3 – ESTIMATING THE ROADWAY GEOMETRY FROM INERTIAL MEASUREMENTS

Position and orientation location systems equipped with Inertial Navigation Systems (INS) coupled with Differential Global Positioning System (DGPS) have become quite popular in a variety of applications including highway evaluation operations. DGPS navigation on its own can accurately identify the location of a moving vehicle using two antennas that receive signals transmitted by four satellites orbiting the earth. On the other hand, a typical INS system contains an Inertial Measurement Unit (IMU) with three gyroscopes and three linear, single-degree, accelerometers installed along the three orthogonal axes to measure, respectively, the 3-D orientation and the acceleration components. It also includes a navigational computer that facilitates the navigation solution through its interaction with the DGPS unit. INS/DGPS integrated systems are widely used in navigation due to the advantages they offer when used as an integrated system as opposed to separate units (Wei 1990, Scherzinger 2000).

The MPSV is also equipped with an INS/DGPS coupled system that can measure the vehicle's spatial location and the orientation at any given instant during the evaluation operation. To accomplish this task the INS/DGPS system uses state-of-the-art fiber optic gyroscopes and silicon accelerometers in its IMU. The integration of the INS and the DGPS is achieved using a statistical algorithm which minimizes the error involved in the

position prediction. This algorithm and many other relevant programs are executed on the POS Computer System (PCS) that controls the above system.

3.1 Scope of Investigation

The roadway geometric data typically obtained by the INS/DGPS coupled system are (1) cross-slopes, (2) grades of vertical curves (3) radii of curvature of horizontal curves, and (4) GPS data. High sensitivity of the instruments in the IMU unit and the relatively high frequency of data gathering (i.e.200 Hz) combined with even small scale variability caused by either the roadway, the operator or the instruments themselves can introduce significant errors in the INS/DGPS readings. This paper reports the results of an experimental program that was executed to compare the INS/DGPS geometric data with the corresponding manual measurements.

3.2 Evaluation of Cross slope

The function of the cross-slope in a roadway is to accelerate the draining of rain water and reduce the hydroplaning potential. The typical cross-slope requirement of a roadway is 2% in order to avoid ponding of water and facilitate runoff from the roadway. In addition, roads are sloped in the lateral direction to control the lateral wandering of vehicles traveling on a bend by meeting the centrifugal force requirement of the vehicle. This construction feature is commonly known as the super-elevation at the bend and is different from the roadway cross-slope in that the former is only provided in the direction of the curve, in all of the lanes. For a curve, the super elevation has to be estimated based on the speed limit of the roadway, the radius of the curve, and friction characteristics (Roadway design manual, MNDOT).

$$e + f = \frac{v^2}{127R} \tag{3-1}$$

Where e is the super-elevation of the roadway in (m/m), f is the side friction factor, v is the travel velocity in (km/h) and R is the radius of curvature of the curve in (m).

When measuring the cross-slope the standard procedure is to measure the height at two different locations on the lane, typically the edge near to shoulder of the roadway and the crown. By dividing this height difference by the lane width, one would obtain the cross-slope of the lane (Figure 16) as:

$$\text{Cross - slope} = \frac{\Delta \text{H}}{\text{L}} = \frac{(y_2 - y_1)}{\text{L}} \tag{3-2}$$

In the FDOT pavement evaluation vehicle, the instruments used in measuring the cross-slope include four laser sensors that are attached to the front bumper and the Inertial Measurement Unit (IMU). The four laser sensors provide distance from the front bumper of the vehicle to the roadway surface. The best-fit straight line of the heights measured by the laser sensors provides the road slope with respect to the vehicle bumper. In addition, the tilt of the vehicle body and the bumper in the lateral direction can be obtained by the IMU, if the IMU is considered to be firmly fixed to the floorboard of the van. Then by considering the difference between the above two readings; (1) the tilt of the vehicle floor and (2) the slope of the road visualized by the lasers, one can approximately determine the cross-slope of the roadway. Figure 16 elaborates the above measurements.

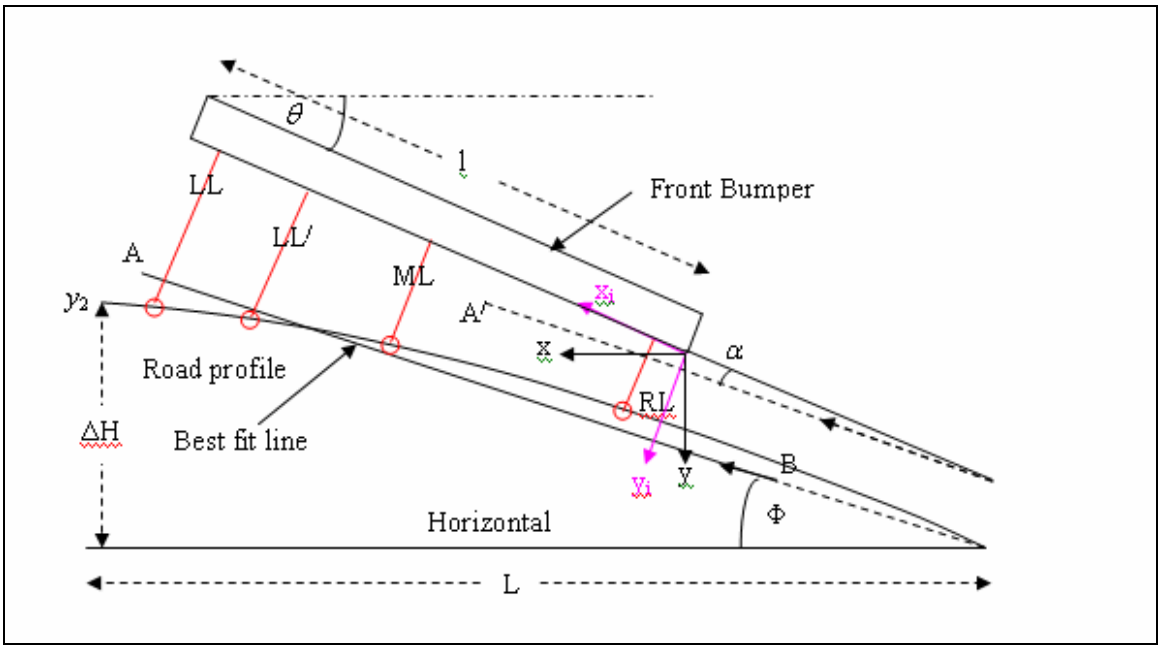


Figure 16: Estimation of cross slope from laser measurement height and IMU roll angle.

Where,

α - Slope measured by the laser sensors with respect to the AB.

Φ- Slope of the best fit line AB.

θ - Roll angle measured by the IMU.

AB- Best fit line for the laser sensor readings.

A/B/- Parallel line to AB drawn at B/.

RL- Laser sensor located at the right corner of the bumper.

ML- Laser sensor located in the middle of the bumper.

LL- Laser sensor located at the left corner of the bumper.

LL/- Laser sensor located in between LL and ML.

l: Width of the vehicle bumper (in inches).

L: Width of the lane (in feet).

(x_i,y_i): Coordinates with respect to the bumper.

y_2, y_1: measured heights at the crown and the shoulder of the lane respectively, expressed in horizontal-vertical (x,y) coordinate system.

The approximate cross slope of the roadway is given by:

$$\Phi = \theta - \alpha \, . \tag{3-3}$$

In this study, alternative methods were also explored in determining the cross-slope of the roadway from the available test data. Basis for the second method in estimating cross-slope was the Least Squares Approximation (LSA) of the laser sensor readings using a parabolic fit. All of the laser measurements obtained from the vehicle are specified with respect to the coordinate system originating from the right corner of the vehicle bumper (Fig 16). Therefore, Eqn. (3-4) is used to convert the laser coordinates from vehicle bumper frame to the vertical and horizontal (x,y) coordinate system shown in Figure 16.

Transformation matrix for the coordinate transformation can be expressed as:

$$\mathbf{t} = \begin{pmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{pmatrix} \tag{3-4}$$

And the (x, y) coordinates are obtained as:

$$\begin{pmatrix} x \\ y \end{pmatrix} = \mathbf{t} \begin{pmatrix} x_i \\ y_i \end{pmatrix} \tag{3-5}$$

3.2.1 Linear L. S. A. Fit (The Existing Algorithm)

Thus the equation of LSA linear fit could be given as:

$$y = a_m x + b_m \tag{3-6}$$

3.2.2 Least Squares Parabolic Fit (The Modified Algorithm)

Since the road profile can be better approximated by a second order curve, a more accurate estimate can be obtained by using a second order (parabolic) fit. In this case, one can use a second order fit as:

$$y = a_m x^2 + b_m x + c_m \tag{3-7}$$

Once the least squares linear or the parabolic fit is obtained, the two curves are then extended to the total length of the road to determine the road cross-slope as measured by the evaluation van. To achieve this it is required to make the assumption that the vehicle travels in the middle of the lane. After the height difference is estimated one can obtain the cross slope from Eqn (3-2).

3.3 Radius of Curvature

The proper design of the radius of curvature of the roadway at a horizontal curve is vital in making the roadway safe for its users at the design speed. Therefore, an important task of pavement infrastructure evaluations is the verification of radius of curvature to be within the acceptable limits at all horizontal curves of a roadway. This especially applies

to rural roads and minor arterials that have been constructed without adhering to appropriate standards. In addition, due to the relationship between the super-elevation and the radius of curvature (Eqn 3-1), knowledge of the radius of curvature at a location is essential in checking the adequacy of super-elevation of that location. This becomes a critical issue when pavement evaluators are called upon to perform safety inspections of specific locations that are known to present safety hazards. Automated and rapid curvature information provided by the IMU system would be immensely helpful in such situations especially from the perspective of safety of survey crews.

Roadways are designed in such a way that when a vehicle moves from a straight section of a road onto a curved section, the vehicle would experience a relatively smooth transition in curvature. This is achieved by constructing transition curves at the extremes of the actual curve. Typically, this transition can be provided by two spiral curves (1) a lead-in, which is the roadway section that starts asymptotic to the straight line segment and merges with the circular curve, and (2) a lead-out, the roadway section which diverges from the circular curve and merges with the second linear segment (Fig 17). In order to achieve this spiral curve, designers use a gradually varying radius of curvature from a straight run to the design circular curvature of the horizontal curve. Also the lead-in and lead-out play a major role in gradually adjusting the super-elevation from the regular cross slope on a straight roadway to the maximum super-elevation at the curve. This is also known as the super-elevation runoff.

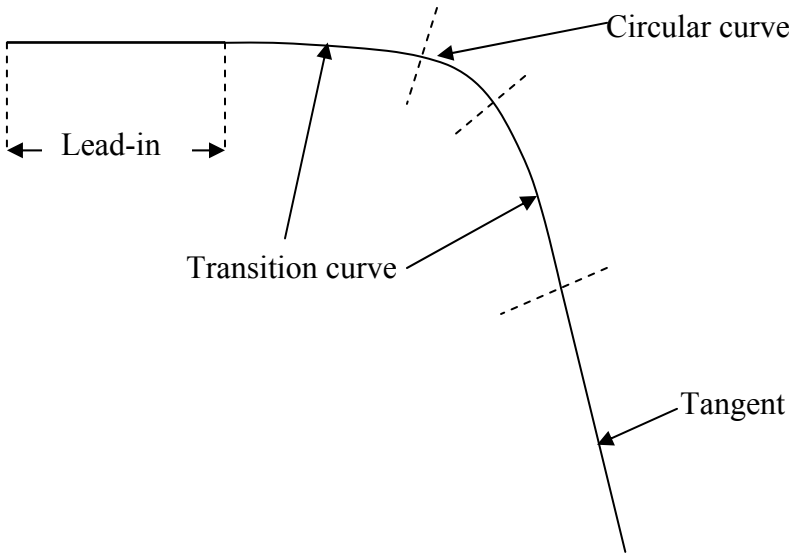


Figure 17: Geometry of the first curve of the reverse curve section.

Generally, two methods can be employed to determine the radius of curvature of a horizontal curve using the IMU readings. They are,

1. The kinematic method using the velocity in the longitudinal direction and the centrifugal acceleration.
2. The geometric means by plotting the actual vehicular horizontal trajectory using the velocity vector and then determining the radius

3.3.1 Determination of Curvature Using Kinematics

The radius of the curvature of the circular section of the horizontal curve can be determined using the following equation of circular motion:

$$r = \frac{v^2}{a} \tag{3-8}$$

Where *v* is the velocity of the vehicle in the longitudinal (body X) direction and *a* is the centrifugal acceleration of the vehicle on a horizontal plane. Since both *v* and *a* are obtained by the IMU, the algorithm used in the FDOT pavement evaluation vehicle uses Eqn (3-8) to determine the instantaneous radius of the horizontal curve.

Since the entire horizontal curve is not an exact circular curve due to the spiral shaped transition curves (lead-in and lead-out) where there is a gradual change of curvature, strictly, the generalized kinematic theory must be used in the determination of the instantaneous radius. Hence, on a non-circular curve, the following relationships hold among the kinematic properties of the vehicle and geometric parameters of the roadway (Beer 1977).

Velocities:

$$v_r = \dot{r} \tag{3-9a}$$

$$v_\theta = r\dot{\gamma} \tag{3-9b}$$

Accelerations:

$$a_r = \ddot{r} - r\dot{\gamma}^2 \tag{3-10a}$$

$$a_\theta = r\ddot{\gamma} + 2\dot{r}\dot{\gamma} \tag{3-10b}$$

Where r, γ are respectively the radial and angular coordinates of the vehicle in a horizontal plane. Within the circular curve since *r* is a constant $\dot{r} = \ddot{r} = 0$. Then it can be shown that the expression for the radius of curvature (*r*) reduces to Eqn (3-8).

3.3.2 Determination of Curvature Using Geometry

The velocity vector obtained from the IMU can be used to plot the trajectory of the vehicle. Because of high frequency of the data collection (200 Hz), the error associated with the linear approximation tends to be relatively low with respect to the distance travelled. The basis of estimating the longitudinal displacement of the vehicle between two consecutive data points *i* and (*i+1*) is illustrated in Eqn (3-11) using the average velocities at those two points. Because the horizontal curve is two dimensional, only the horizontal velocities must be considered.

$$ds = \frac{1}{2}(v_i + v_{i+1})dt \tag{3-11}$$

Where ds is the longitudinal displacement that occurs when the vehicle travels between the points *i* and *(i+1)*, v_i and $v_{(i+1)}$ are the respective velocities at points *i* and *(i+1)* and the dt represents the time elapsed during the travel between points *i* and (*i+1*).

If the vehicle trajectory on the x-y plane is described by the function y=f(x), the radius of curvature can be evaluated by the following equation;

$$R = \frac{\left(1 + \left(\frac{dy_i}{dx_i}\right)^2\right)^{3/2}}{\left(\frac{d^2 y_i}{dx_i^2}\right)} \tag{3-12}$$

Since x and y coordinates are found in terms of time as discrete quantities, the second order Forward Difference formulae can be used to numerically evaluate Eqn (3-12).

$$x^{/} = \frac{(-x_{(i+2)} + 4x_{(i+1)} - 3x_i)}{2h} \tag{3-13}$$

$$x^{//} = \frac{\left[-x_{(i+3)} + 4x_{(i+2)} - 5x_{(i+1)} + 2x_i\right]}{h^2} \tag{3-14}$$

Where, $h = t_{(i+1)} - t_i$ and i is the index that represents all gathered data points. Similarly expressions can be written for $y^{/}$ and $y^{//}$ as well. The numerical forms of the derivatives needed for Eqn (3-12) can be derived as:

$$\frac{dy_i}{dx_i} = \frac{y^{/}}{x^{/}} \tag{3-15}$$

$$\frac{d^2 y_i}{dx_i^2} = \frac{[y^{//}x^{/} - x^{//}y^{/}]}{[x^{/}]^2} \tag{3-16}$$

3.4 Estimation of Curvature in the Field

Two common methods of estimating the radius of curvature of a horizontal curve used by accident investigators are given in (Glenon 2003). Both of these methods yield accurate results when used on a circular section. Due to the ease of measurement and minimal labor requirement, these methods are widely used in the field. The two methods are outlined below:

3.4.1 Chord-Offset Method

This method uses a 100-foot tape and a carpenter's ruler to determine the radius of curvature. The 100-foot tape is placed on either end at the precise edge of the roadway so

that an arc is separated from the curve. Then the carpenter's ruler is used to measure the distance between the mid-point of the 100-foot tape and the edge of the roadway perpendicularly across from the mid point of the 100-foot tape (offset). Once these two measurements are obtained, the radius of the curvature can be estimated by the following geometrical relationship:

$$R = \frac{b^2}{8m} + \frac{m}{2} \tag{3-17}$$

Where,

R is the radius of the curvature.

b is the chord length.

m is the measured middle offset.

3.4.2 Compass Method

This method uses the distance traveled along a segment of the circular section, or the arc length, and the angle subtended by that segment of the curve on the center of the circle (included angle) to estimate the radius of curvature. Arc length can be measured by a measuring wheel or the vehicle DMI. A compass can be used to find the included angle, since it is equal to the deviation between the tangents to the circular segment at its extremes. Once the arc length and the included angle are available, the radius of the curvature can be estimated by,

$$R = \frac{s}{\phi} \tag{3-18}$$

Where,

R is the radius of the curvature of the horizontal curve.

s is the arc length of the considered circular segment.

ϕ is the included angle measured in radians.

In order to address the undulations of the predicted radius values, the same data are first averaged and moving average, moving median is used to respectively smoothen out and eliminate the outliers. Averaging has been performed by considering a pre-determined number (n) of consecutive data points at one time, according to Eqn (3-19).

$$p_{avg} = \frac{(p_1 + p_2 + p_3 \cdots + p_n)}{n} \tag{3-19}$$

Where, *p* denotes the variable that needs to be averaged. The average values for points (x,y) obtained from the operation of Eqn (3-19) are then used as the new x and y coordinates for the averaged time period (i = 1…..n). The interval size chosen in this work 10. Subsequently, the second order forward difference algorithm (Eqns. 3-13 –3.14) was applied to the x and y coordinates to find the first and second derivatives needed for Eqn (3-12).

3.4.3 Manual Estimation of Radius of Curvature

The radius of curvature was also estimated using a manual survey the results of which are given in Table 3.

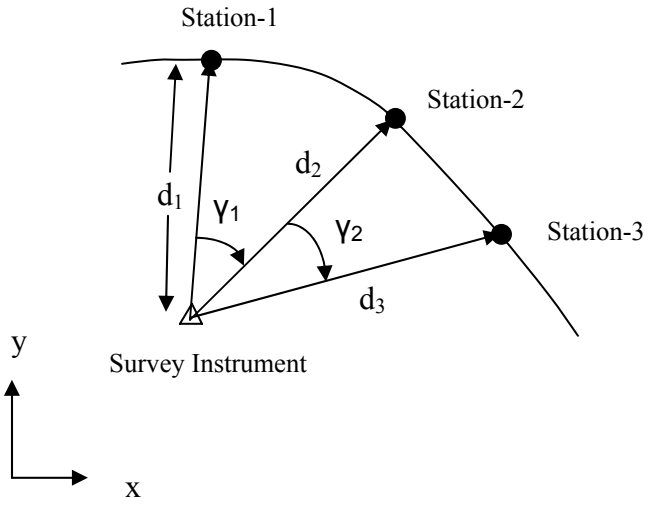


Figure 18: Illustration of notation used in Table 4.

Table 3: Data obtained from the manual survey.

Distance(d) (ft)	Horizontal angle (γ)(deg)	X (ft)	Y (ft)
12.89	0	12.8900	0
28.36	58.37611	14.8703	-24.1488
46.33	68.54306	16.9476	-43.119
66.39	72.21167	20.2822	-63.216
86.58	73.74528	24.2344	-83.1191
106.36	74.20806	28.9453	-102.346
126.57	74.25056	34.3550	-121.818
161.37	73.59861	45.5652	-154.803
205.11	72.77528	60.7372	-195.911
225.29	72.7325	69.2826	-215.136
244.66	72.96361	72.4964	-233.672
264.86	73.38167	75.7486	-253.797
284.17	73.92778	78.6716	-273.063

CHAPTER 4 – FUNDAMENTALS OF VISION NAVIGATION

4.1 Introduction

Utilization of a sequence of images obtained by a camera fixed on a moving vehicle in order to estimate the navigation parameters of the vehicle is the primary concern of this chapter. Estimation of navigation parameters, specifically rotations and translations of the camera with respect to the location of the camera at the first image point, comes under a special area in Computer Engineering named Computer vision. This specialization originated with the revolutionary ideas of psychologist David Marr in application of psychology and neurophysiology in visual processing. This application, machine vision, attempts to setup an analogy between human or biological vision systems and images obtained using a camera.

Machine vision deals with estimating the motion of a camera evaluated using a sequence of captured images. In understanding how one could extract useful navigation information from a sequence of images it is important to grasp the fundamentals of how an image is created in a camera. In addition, to the acquisition of an image, it is important to understand how a camera works and how one can model this computationally. The simplest and the most fundamental model for image formation in a camera is the *pinhole* camera model. The pinhole camera model consists of a small hole which allows the light rays coming from a 3D point travel through it and form an inverted object on the image

plane. For simplifying the visualization of image formation this image plane can be placed in between the principal point, the small hole in pinhole model, and the object. This can be elaborated using Fig. 19.

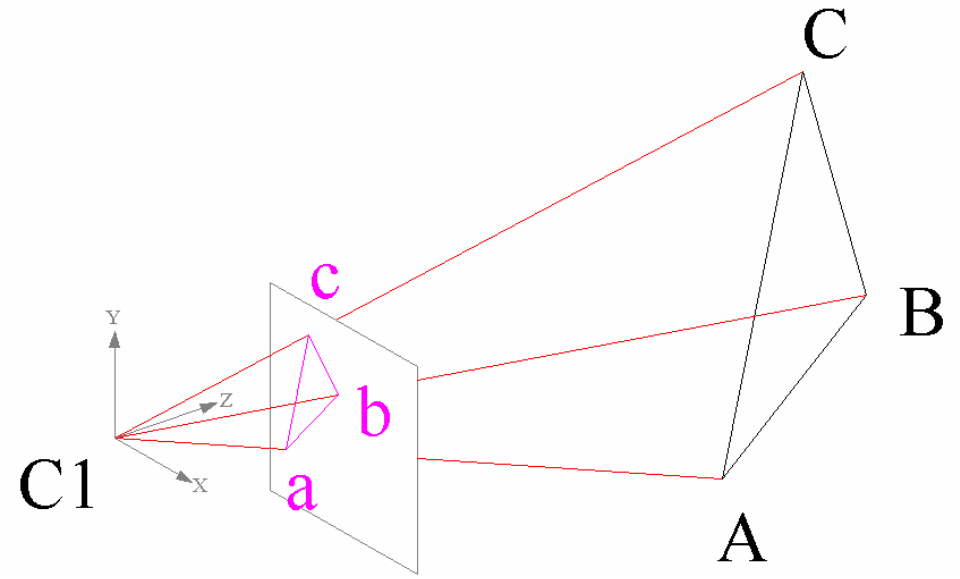


Figure 19: Image formation in a pinhole camera.

Where, ABC represents an object on 3D space while abc is the corresponding image point of the 3D points on the image plane. The small hole, pinhole, is denoted by C1. Note that the image plane in Fig. 19 is represented by a hypothetical plane in front of the pinhole to avoid the undesirable property of inverting the object on the image plane. This not only helps visualize the image formation easily but also simplifies the image analysis process. It can be seen from Fig. 19 that image formation creates a 2D image from 3D points, i.e. image formation process is nothing but a transformation from the 3D space on to the 2D space by using straight lines that pass through a single point, the principal point. This is also called the perspective projection. In fact Euclidean geometry is a special case of projective geometry and the application of projective geometry in machine

vision makes the computational task simpler. The relationship between the 3D coordinates and the 2D coordinates in Fig 19 can be denoted as,

$$u = \frac{f}{z} x \tag{4-1a}$$

$$v = \frac{f}{z} y \tag{4-1b}$$

Where, (u, v) are image plane coordinates and (x, y, z) are 3D coordinates. The focal length of the camera is given as f. When the Cartesian coordinates are used in order to represent the coordinates on the projected space, Eqns (4-1 a, b), these representations become non-linear. This non-linearity in representation makes the computational task more complex. Therefore, a separate coordinate system, called Homogeneous coordinate system, is introduced to make this non-linearity a linear representation.

4.1.1 Homogeneous Coordinate System

The primary use of the Homogeneous coordinate system in projective geometry is to avoid the non-linearity that is cause when using the Cartesian coordinate system. In addition, Homogeneous coordinates are preferred in computer vision since they represent the translations that occur in between the image locations as a matrix operation. To translate Homogeneous coordinates (x, y, 1) by 2x1 vector **A** (= $[a_1, a_2]^T$)**,** it has to be multiplied by,

$$\mathbf{T} = \begin{pmatrix} 1 & 0 & a_1 \\ 0 & 1 & a_2 \\ 0 & 0 & 1 \end{pmatrix} \tag{4-2}$$

Homogeneous coordinates of an N dimensional projective space can be represented by a vector of length (N+1) where not all the components are zero. For instance, a 3D Homogeneous coordinate system can be expressed by a vector $(x, y, z, w)^T$ where at least one of the components is non zero. Application of linear algebra to the Homogeneous coordinate system becomes complicated as one given point on that system is obtained by multiplying one point with a nonzero scale factor. That is, $\mathbf{x} = \lambda\mathbf{y}$; for all $\lambda \neq 0$, where **x** and **y** are homogeneous coordinates which represent the same point. The basis of an N-D projective space contains (N+2) points such that there are (N+1) of them are independent.

If one compares the Homogeneous coordinates and Cartesian coordinates one can see that for every value of w other than w = 0, Homogeneous coordinates can be represented by an equivalent Cartesian coordinate point given by $\left(x/w \quad y/w \quad z/w \quad 1 \right)^T$. When, w = 0, it is impossible to represent in Cartesian coordinates and this plane is identified as the plane of infinity on Homogeneous coordinate system. This plane makes the Homogeneous coordinate system a compact space making it distinguishable from Cartesian coordinate system. This property solely makes the Homogeneous coordinate system more suitable in computer vision technology.

4.2 Estimating Intrinsic Parameters of the Camera

In revising the IMU data with respect to its inherent error accumulation problem one needs to obtain the position and orientation at each camera location. Since the camera is rigidly fixed to the vehicle, these camera positions and orientations can be considered as the vehicle's navigation parameters as well. However, it is essential that the camera be first calibrated for its intrinsic parameters, such as the focal length, principal point and distortion parameters. Furthermore, the fixed transformation between the *b-frame* and the *c-frame* must be determined.

Intrinsic properties of the camera involve six parameters; *focal length, scale factor, radial distortion coefficients* (2 unknowns for a second order approximation), and *tangential distortion coefficients* (2 unknowns). In this work the algorithm introduced by Heikkila et al. (Heikkila 2000) is used to estimate the intrinsic properties of the camera. The basic models used for the camera and the distortion parameters (Heikkila 2000) are summarized below.

The perspective camera model which is used in homogeneous coordinate systems can be expressed as:

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \mathbf{F} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \mathbf{P}.\mathbf{M}. \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{4-3}$$

Where the **P** matrix denotes the intrinsic properties, the **M** matrix denotes the extrinsic properties related to rotation and transformation and **F** is the combination of **P** and **M** known as the perspective transformation matrix. u and v are the measured pixel coordinates obtained from the image. **P** and **M** can be expressed as:

$$\mathbf{P} = \begin{bmatrix} sf & 0 & u_0 & 0 \\ 0 & f & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{4-4}$$

$$\mathbf{M} = \begin{bmatrix} \mathbf{R} & \mathbf{T} \\ 0 & 1 \end{bmatrix} \tag{4-5}$$

Where s is the aspect ratio, f is the focal length and (u_0, v_0) are the coordinates of the principal point on the image plane. **R** and **T** denote the rotational and translational matrices between the two frames, i.e. from the camera frame to the global frame. The camera lens introduces nonlinear distortions due to the fact that the lens system is composed of several optical elements. Therefore, the perspective projection is insufficient for accurate modeling without the incorporation of the magnitude of distortion. The distortion is divided into radial and tangential (de-centering) components. The second order radial distortion model used in this work can be expressed as:

$$\mathbf{D}_r = \begin{bmatrix} \bar{u}_d (k_1 r_d^2 + k_2 r_d^4) \\ \bar{v}_d (k_1 r_d^2 + k_2 r_d^4) \end{bmatrix} \tag{4-6}$$

Where, k_1, k_2 are coefficients that express the radial distortion, $\bar{u}_d = u_d - u_0$, $\bar{v}_d = v_d - v_0$ $r_d = \sqrt{\bar{u}_d^2 + \bar{v}_d^2}$ and (u_d,v_d) are the distorted coordinates of any point of the image. Herein

only the first two terms of the radial distortion model Eqn (4-6) are considered due to their dominant role in describing the radial distortion and also to avoid any numerical instability that would be caused by a more complicated model (Zhang 2006, Heikkila 2000).

Similarly the tangential distortion model is given as;

$$\mathbf{D}_t = \begin{bmatrix} 2p_1\bar{u}_d\bar{v}_d + p_2(r_d^2 + 2\bar{u}_d^2) \\ p_1(r_d^2 + 2\bar{v}_d^2) + 2p_2\bar{u}_d\bar{v}_d \end{bmatrix} \tag{4-7}$$

Where, p_1, p_2 are coefficients that express the tangential distortion. Therefore, the corrected coordinates of a point (u_c, v_c) on the image can be expressed as;

$$\begin{bmatrix} u_c \\ v_c \end{bmatrix} = \begin{bmatrix} u_d \\ v_d \end{bmatrix} + \mathbf{D}_r + \mathbf{D}_t \tag{4-8}$$

The process of estimating the intrinsic and extrinsic properties of the camera is initiated by forming the initial **P** and **F** matrices from the nominal parameter values and the perspective model respectively. Using initial **P**, **F** and the orthogonality constraint ($\mathbf{R}^T\mathbf{R}=\mathbf{I}$) of the rotation matrix, the initial **M** can be determined. Once the initial estimates are available, the upgraded estimates of intrinsic parameters can be obtained by minimizing the weighted sum of squares of error between the observations and the model.

4.3 Feature Corresponding in Images

The camera is calibrated for the intrinsic and extrinsic parameters in order to use it in estimation of motion parameters from a sequence of images obtained from it. The entire motion estimation problem is two folds. They are,

1) Correspondence problem.
2) Motions estimation problem.

The two following sections address the motion estimation problem for each type of correspondences that are used in this work. Estimation of motion from a sequence of images is performed by establishing common objects that are visible in consecutive images. These common objects could be of any geometry, such as points or pixels, straight lines etc., and are called in this dissertation as features, deviating a little bit from the definition of features in computer vision literature. The same features on two or more images are called correspondences making the entire process of defining and locating features, feature correspondence. In this dissertation two types of features are considered, namely points and straight lines. Also two different methods of correspondences are utilized,

1) Manual feature correspondence.
2) Automatic feature correspondence.

The typical point feature, or pixel, correspondences are any point on the image that can be reliably distinguished from the rest of the points and easily located on the very next image. Similarly, the straight lines must also be reliably recognized and located on the next two images. As it is understood from the definition of the feature and reliable

tracking constraints there is a certain ambiguity involved in the correspondence problem. Some of the most common and established methods of identifying point correspondences are,

1) Light reflected by any object or point.

   This method involves measuring the irradiance from a point and locating the same point on the second image that has the same irradiance value. Since in the nature surfaces are in between matte (reflecting light in all directions at the same intensity) and glossy (acting totally as a mirror), locating the exact point that has the same illumination is quite difficult.

2) Identifying edge pixels.

   This method involves establishing pixels that are on the edge of any object which is distinguishable from the rest at ease. This can be used to pick up the correspondences of both types, i.e. points and straight lines. In this work the feature correspondences are picked according to this method.

In addition, to reduce the ambiguity involved in establishing the correspondence problem one can use some constraints. One of the most common constraints in establishing accurate point correspondences in two images is drawing the epipolar line. Establishing and drawing the epipolar lines and planes is described in depth in the next section. Furthermore, one can use geometric constraints arising from the object in order to obtain more accurate feature correspondences. In establishing the point correspondences in this research work the authors utilized a well established KLT (Kanade-Lucas-Tomasi)

feature tracker. This program is completely automated and the point features are tracked in the sequence of images with replacement (Fig. 20). Of these feature correspondences only the ones that are tracked in more than five images are identified and used as an input to the motion estimation algorithm for estimating the rotation and translation. On the other hand, straight line features of any given image were extracted, first using an edge detector program (Fig. 21-a), the Canny edge detector (Canny 1984), to identify the object boundaries or edges. Then the correspondence of lines in a sequence of images is identified manually (Fig. 21-b). This method can be implemented more conveniently on highway travel because at highway speeds the correspondence of line features in a sequence of images can be achieved easily and more accurately than the correspondence of point features.

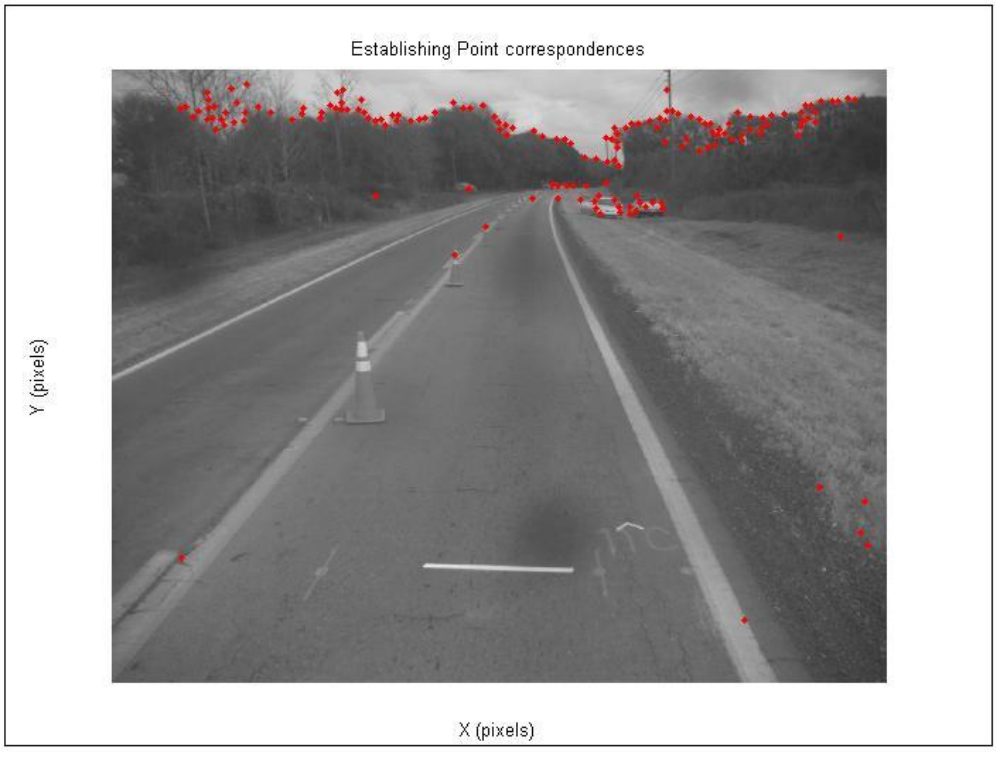
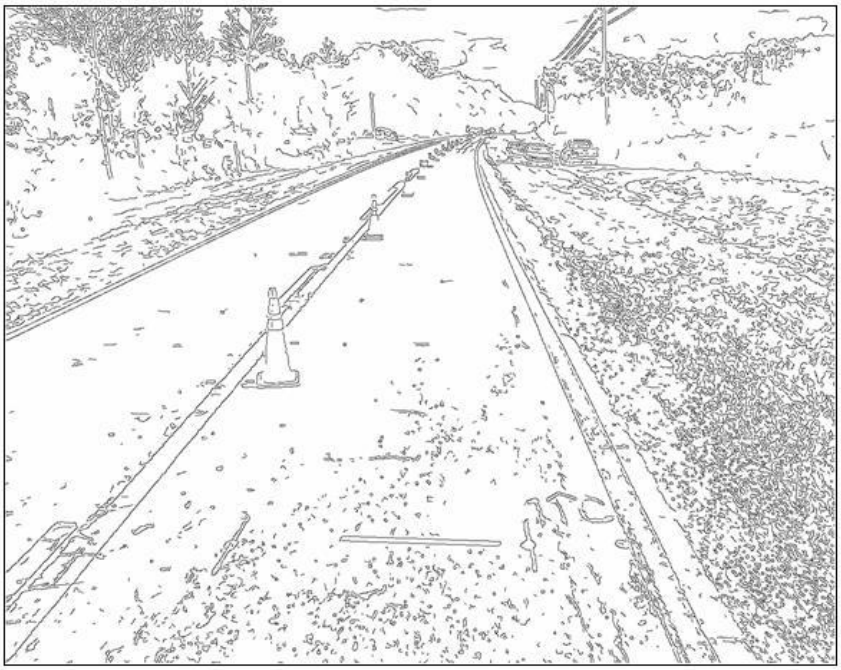
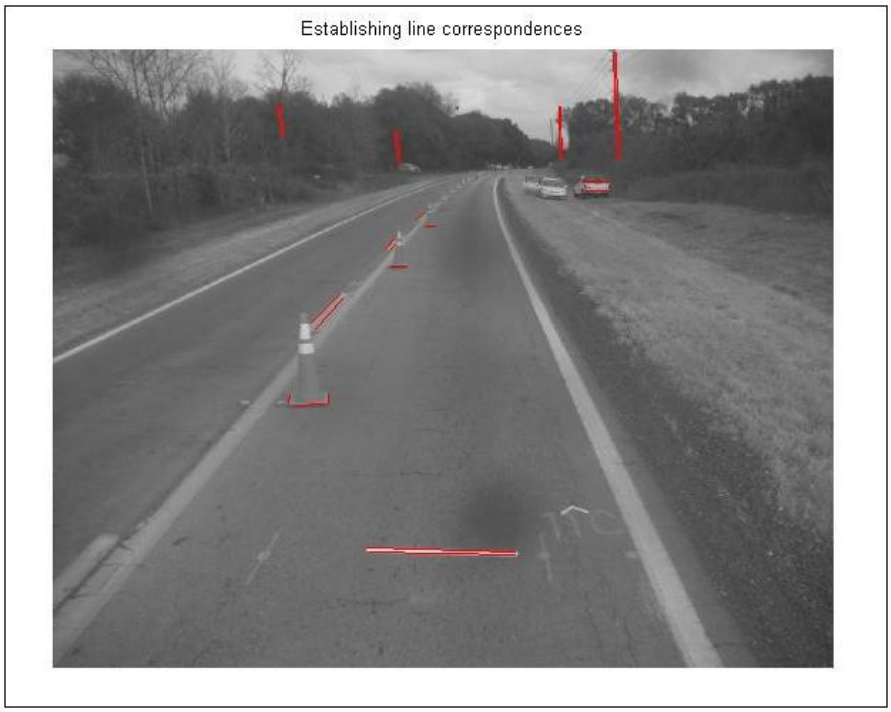


Figure 20: Established point correspondences from KLT tracker.



(a)



(b)

Figure 21: (a) Image after edge detection, (b) Establishing correspondences.

When the feature correspondences are picked one must be careful not to pick features that are moving with respect to the camera. This is due to the fact that the motion estimation algorithm considers the features to be stationary while the camera is in motion. One other important fact in picking line correspondences is not to pick all the lines on the same plane which would cause degeneracy in the motion estimation algorithm. In addition, when establishing correspondences feature occlusion, i.e. depth discontinuities where the normal vector of the feature is away from the camera, must be avoided.

4.4 Estimating Motion from Point Correspondences

Once the point features are tracked in the sequence of images they are used to estimate the rotation and translation of the camera between two consecutive images. The algorithm used to estimate the pose requires at least eight point correspondences that are non-coplanar to solve for the pose of the camera in two consecutive images. Therefore, the algorithm is called eight-point algorithm. A description of eight-point algorithm follows.

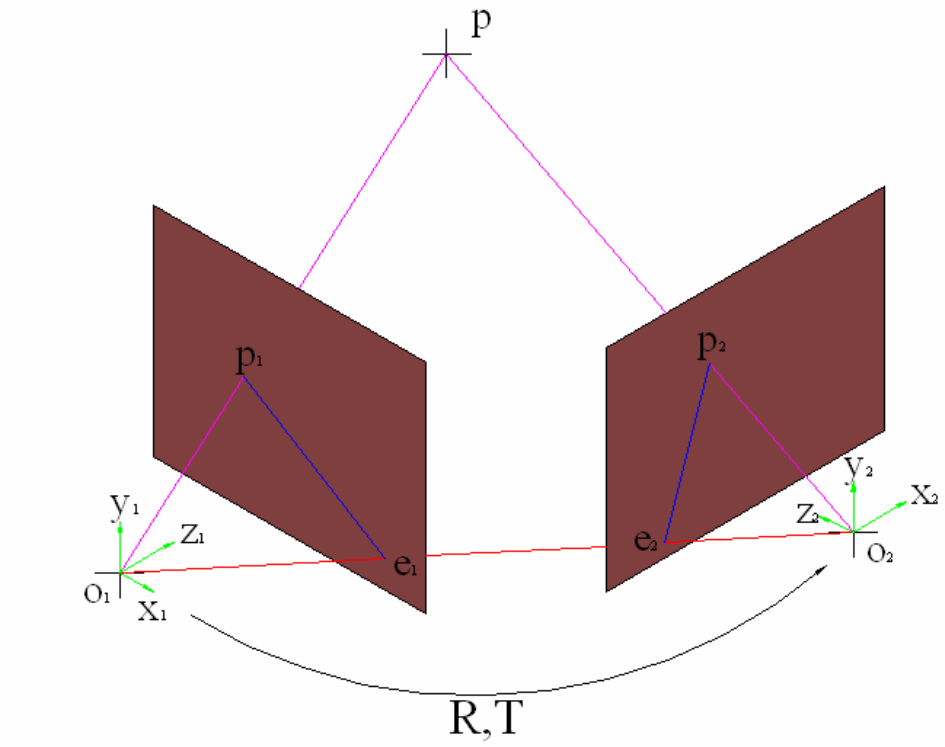


Figure 22: Rigidity constrain for estimating *pose* from point correspondences.

Fig. 22 shows two images captured by the camera at two consecutive time instances. Point p is a point common to both images and o_1 and o_2 are the cameras coordinate frame centers at two instances. Point p_1 and p_2 are respectively the projection of the 3D point p on to two image planes.

The *epipole* points where the line joining two coordinate centers and the two image planes intersect are denoted by e_1 and e_2 respectively while the line joining e_1p_1 and e_2p_2 are called *epipolar lines*. Such epipolar lines pass through a common point (*epipole*) which corresponds to the image of the center of the camera at the first position, on the second image plane. Then the second point of any correspondence pair (such as a point A

in Fig. 23) must lie on the epipolar line (BC in Fig. 23) obtained from its partner point in the first image with the *epipole*. This is only applicable in situations where the locus of the *c-frame* center is not parallel to any of the consecutive image planes considered, a condition which is generally satisfied in automated driving tasks. Thus the importance of an epipolar line lies in that it restricts the placement of any correspondence point on the second image to a straight line obtained using its partner point in the first image.

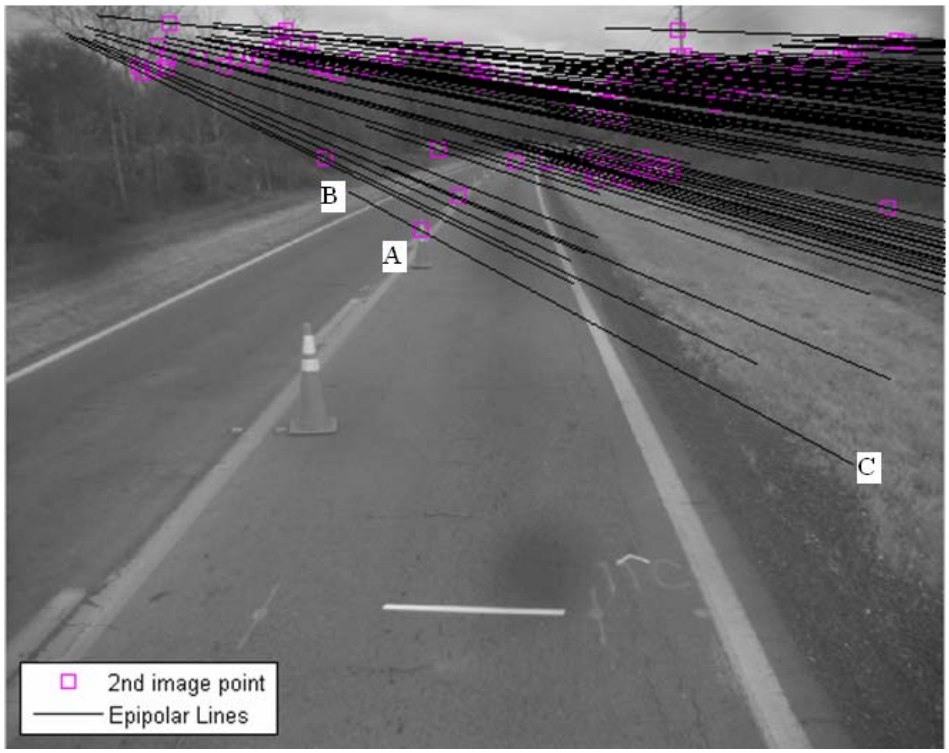


Figure 23: Epipolar lines drawn for the vision system.

Let the rotation and translation between the two images are denoted as R and T. If the coordinates of points p_1 and p_2 are denoted as (x_1, y_1, z_1) and (x_2, y_2, z_2) respectively then the two coordinates can be related as,

$$\begin{pmatrix} x_2 & y_2 & z_2 \end{pmatrix}^T = R\begin{pmatrix} x_1 & y_1 & z_1 \end{pmatrix}^T + T \tag{4-9}$$

From Fig. 24-a, it is clear that the lines connecting the 3D point p with the camera centers and the line connecting to the two centers are on the same plane. This constraint, which is geometrically shown in Fig. 22, is given in the algebraic form (Faugeras 1996) in Eqn.(4-10). Since the three vectors lie on the same plane,

$$p_1^T \bullet (T \times R p_2) = 0 \tag{4-10}$$

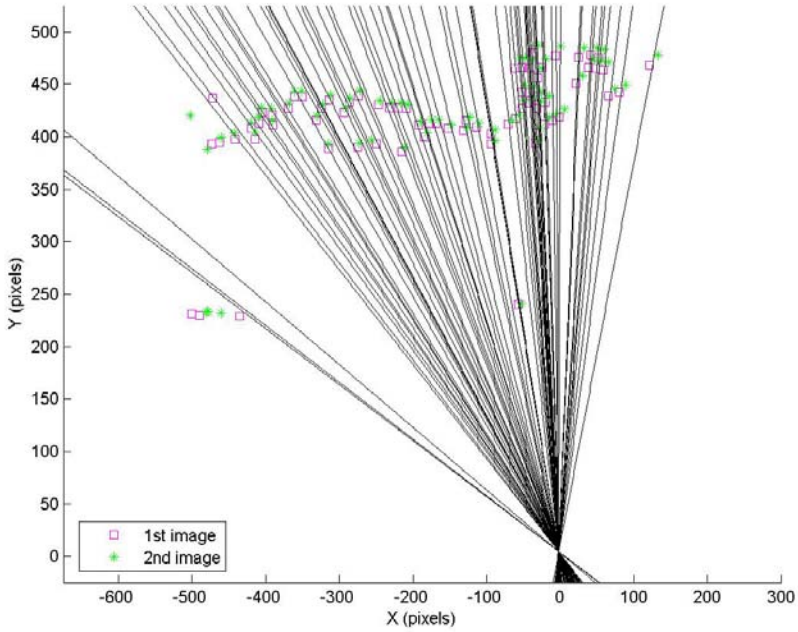
Where, p_1 and p_2 on Eqn.(4-10) are the homogeneous coordinates of the projection of 3D point onto two image planes respectively. Both $T, R\left(\in \Re^3\right)$ are in 3D space, so there will be nine unknowns involved in Eqn.(4-10). But since all the measurements obtained from a camera is scaled in depth one has to solve only eight unknowns in Eqn.(4-10). Therefore, in order to find a solution to Eqn.(4-10) one should meet the criterion, $\text{Rank}(T \times R) \geq 8$ .Let $E = T \times R$ the unknowns in E are taken as (e_1 e_2 e_3 e_4 e_5 e_6 e_7 e_8) and the scaled parameter is taken to be 1. Then one could setup Eqn.(4-10) as,

$$A\vec{e} = 0 \tag{4-11}$$

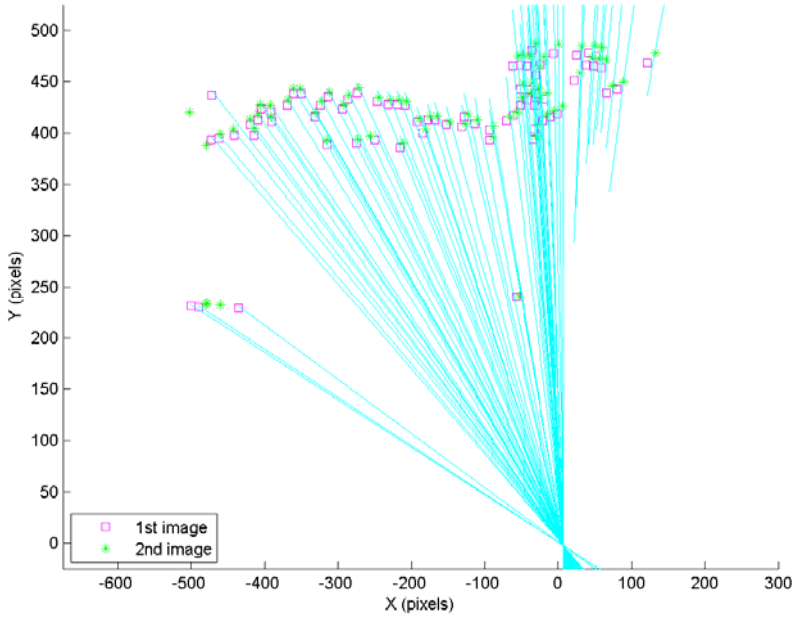
Where $A = \left(x_1x_2 \quad x_1y_2 \quad x_1f \quad y_1x_2 \quad y_1y_2 \quad y_1f \quad fx_2 \quad fy_2 \quad f^2\right)$, is known and the unknowns are $\vec{e} = \left(1 \quad e_1 \quad e_2 \quad e_3 \quad e_4 \quad e_5 \quad e_6 \quad e_7 \quad e_8\right)$. Once the sufficient number of correspondence points is obtained Eqn.(4-11) can be solved and the E estimated. Based on the Epipolar condition, the camera centers at each camera location o_1 and o_2 and the images of the point p on first and second image planes p_1, p_2 are on the same plane. Then one can restrict the search space for the second corresponding point on the second image plane to a 1D line. This 1D line search space is given as e_2p_2, epipolar line on the second image plane corresponding to the point p in Fig. 22. In other words, if one knows the

essential matrix, E, and the correspondences of 3D point p on the first image plane the correspondences for the same 3D point p on the second image plane must lie on the epipolar line e_2p_2. This is true for all the correspondences that are picked on both images. The epipolar lines drawn for all the correspondences on the second image plane can be given as;

In estimating the essential matrix (E) using the data obtained from the vision system, i.e. the correspondence points obtained by the tracking algorithm, the only variable involved in the estimation process is the focal length of the camera. Although the essential matrix can be estimated using the Eqn (4-11), the value obtained here could be different from the actual focal length value due to erroneous correspondences in consecutive images. This is evident from the epipolar lines plot given in Fig. 24-a. The reason for this erroneous epipolar lines is because the false correspondences make the estimated Essential matrix erroneous in Eqn. (4-11). Therefore, by removing the false correspondences it can be shown that the epipolar lines also become accurate (Fig. 24-b). In this work the false correspondences have been removed manually.



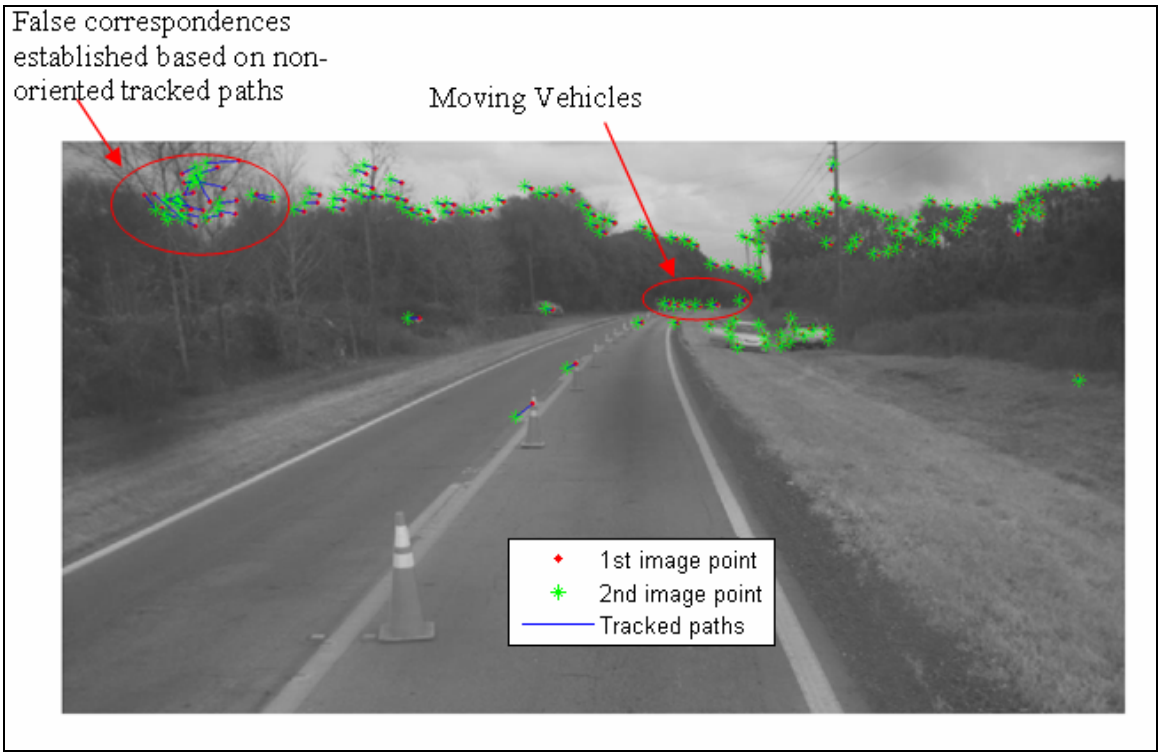
(a)



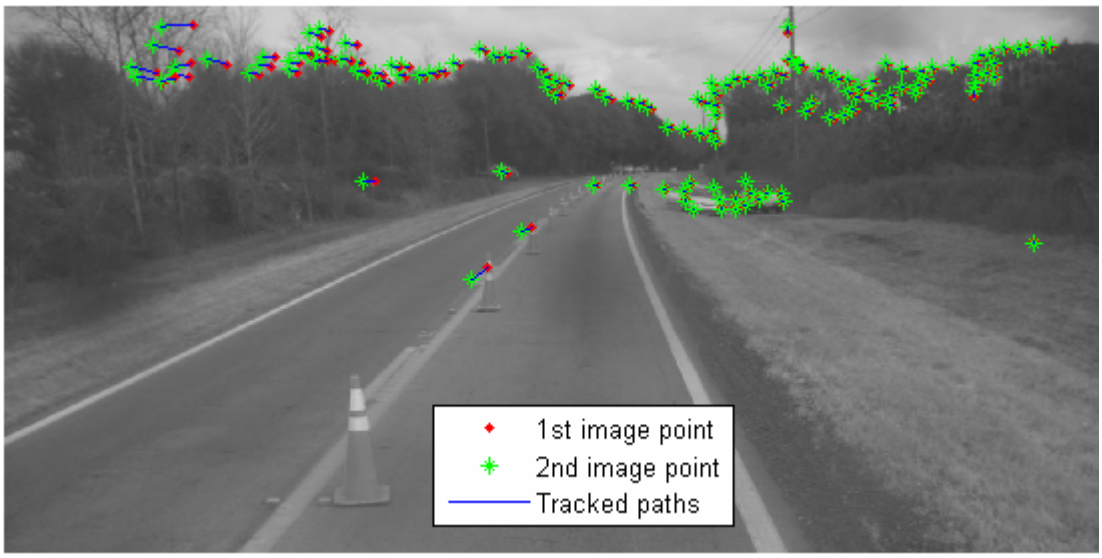
(b)

Figure 24: Epipolar lines (a) estimated before removing false correspondences, (b) after removing false correspondences.

In locating false correspondences before obtaining the accurate Essential matrix, the flow direction of correspondences was used in this work. Locating the false correspondences by observing flow directions of tracked points in two consecutive images involve checking the flow pattern of each tracked path and removing the correspondence points that are randomly oriented. If the features are extracted in two consecutive images from the same location of a stationary object their flow direction must align with the direction that corresponds to the travel direction of the vehicle. As the data for this work was collected in an outdoor uncontrolled environment, movement of the objects used in extracting the features between the image capturing instances, changes of irradiance of the object etc. could cause this subtle errors in extracting correspondences. This is illustrated in Fig. 25-a, b.



(a)



(b)

Figure 25: Filtering point correspondences using motion tracking (a) before (b) after.

From Fig. 25 (a)-(b) one can see how the correspondence flow direction or the tracked paths can be utilized in removing erroneous features from the sequence of images. This

method can be used in removing both stationary and moving features tracked on two consecutive images. Once the erroneous feature correspondences are taken off, the features that are tracked in more than five images are identified and subsequently used as input to the motion estimation algorithm to obtain the vehicle's rotation and translation between each of the frames.

Since, $E = T \times R$ once the matrix E is estimated it can be utilized to recover translations and rotations. The translations (T) and rotations (R) can be given as,

$$\begin{aligned} T &= c_1 \times c_2 \\ (T \bullet T)R &= E^{*T} - T \times E \end{aligned} \tag{4-12}$$

Where, $c_i = T \times r_i$ (for i=1, 2, 3) and the column vectors of R matrix are given as r. Also E^* is the matrix of cofactors of matrix E. In this work, in order to estimate the rotation and translation from given correspondence, the specific algorithm outlined in (Faugeras 1996) was used.

4.5 Estimating Motion from Line Correspondences

From the edges located from an image using the edge detection algorithm one can estimate the *pose* of the camera. Since the line correspondences between the images are used in establishing motion parameters of the camera the problem at hand would become more difficult than the motion estimation using the point correspondences. This could be attributed to the difficulty in constraining the rigid displacement between two consecutive images that have captured the same 3D line segment. In other words, the same 3D line

can be detected on the two consecutive images. However, mere observation of the line segments will not constrain the relative movement of the camera between two image positions. Therefore, the estimation process becomes much more rigorous.

In order to estimate the *pose* of the camera using line correspondences one has to constrain the rigid motion of the camera so that it is possible to obtain an algebraic relationship for the motion. This can be achieved by considering the fact that the intersection of three planes is a line. In physical terms, this says that if one could find a 3D line, *P*, which can be found on three consecutive images then the three planes made by each image's camera center, C_i where i=1, 2 and 3, and the 3D line, *P*, intersect on one straight line which in this case is the 3D line *P*. This is shown schematically in Fig. 26.

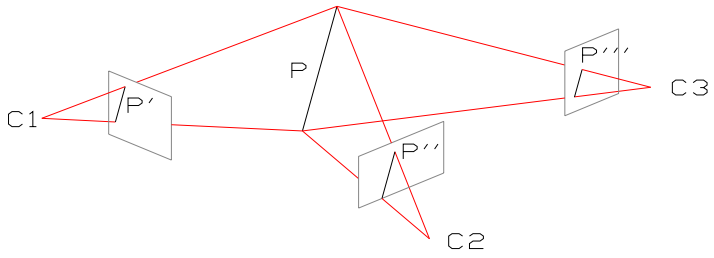


Figure 26: Rigidity constrain for the motion estimation using line correspondences.

Where, P denotes the 3D line and $P^{/}$, $P^{//}$ and $P^{///}$ represent the projections of the 3D line onto the respective image planes. C_1, C_2 and C_3 are the three camera coordinate centers at each image locations respectively. If the three normal vectors to each of the planes given

by $C_1P^{/}$, $C_2P^{//}$ and $C_3P^{///}$ are given as $n^{/}$, $n^{//}$ and $n^{///}$ the three vectors, with respect to the coordinate system at the first camera frame ($P^{/}$, $P^{//}$ and $P^{///}$) can be expressed as;

$$P' = \begin{pmatrix} n' \\ 0 \end{pmatrix} \tag{4-13}$$

$$P'' = \begin{pmatrix} R_1 n' \\ -T_1^T R_1 n' \end{pmatrix} \tag{4-14}$$

$$P''' = \begin{pmatrix} R_2 R_1 n' \\ -(T_1 + R_1 T_2)^T R_2 R_1 n'' \end{pmatrix} \tag{4-15}$$

Once these vectors are obtained one can setup a 4x3 matrix [$P^{/}$, $P^{//}$, $P^{///}$] which can be used to solve for the rotations (R) and translations (T) using the aforementioned rigidity constrain (Eqn(4-16)). The rotations can be found by minimizing the function,

$$\sum_{i=1}^{N} \left(n_i'^T \left(R_1 n_i'' \times R_2 R_1 n_i''' \right) \right) \tag{4-16}$$

Where, N is the number of correspondences detected between the three images. Once the two rotations are obtained the translations can be estimated by minimizing Eqn.(4-17);

$$\sum_{i=1}^{N} \left(\left\| T_1^T R_1 n_i'' \right\| \left\| n_i' \times R_2 R_1 n_i''' \right\| - \left| (T_1 + R_1 T_2)^T R_2 R_1 n''' \right\| \left\| n_i' \times R_2 R_1 n_i'' \right\| \right)^2 \tag{4-17}$$

The solution for the rotations and translations between the images 1, 2 and 3 can be estimated using the Eqn. (4-13) - (4-17). In this work the pose from line correspondences are estimated using the algorithm presented in (Taylor et al., 1996). Appendix A provides the details of this algorithm.

CHAPTER 5 – KALMAN FILTERING

5.1 Introduction

The inherent error growth problem of IMU has to be overcome in land navigation vehicles to obtain a meaningful and reliable navigation solution. In order to minimize the increasing error in IMU measurements the IMU readings have to be updated by an alternative measurement at regular intervals. Since every measurement instrument has errors involved in its measurements, such as bias errors, random measurement errors etc., and have different measurement sensitivities, care must be taken before a measurement from another sensor is utilized in updating a sensitive instrument like IMU. Therefore, to overcome this problem one needs to utilize a fusion technique that would reduce the error involved in both the measurement systems in a statistically optimized manner.

Although there are varieties of statistical filtering techniques, in this work the authors selected the widely applied filtering technique of Kalman filtering. A Kalman filter is a minimum variance estimator designed for linear systems. Therefore, for linear systems Kalman filtering provides the statistically optimized solution. One key reason to select Kalman filtering for fusion of vision sensor data with IMU data was due to its proven accuracy in similar data fusion applications and its wide applicability in the fusion field. In addition, the system equations derived for IMU error dynamics takes a linear form with respect to error terms which motivates the use of the Kalman filtering technique in

the aforementioned problem. All the processes involved in the IMU/Vision fusion considered in this work is illustrated in Fig. 27.

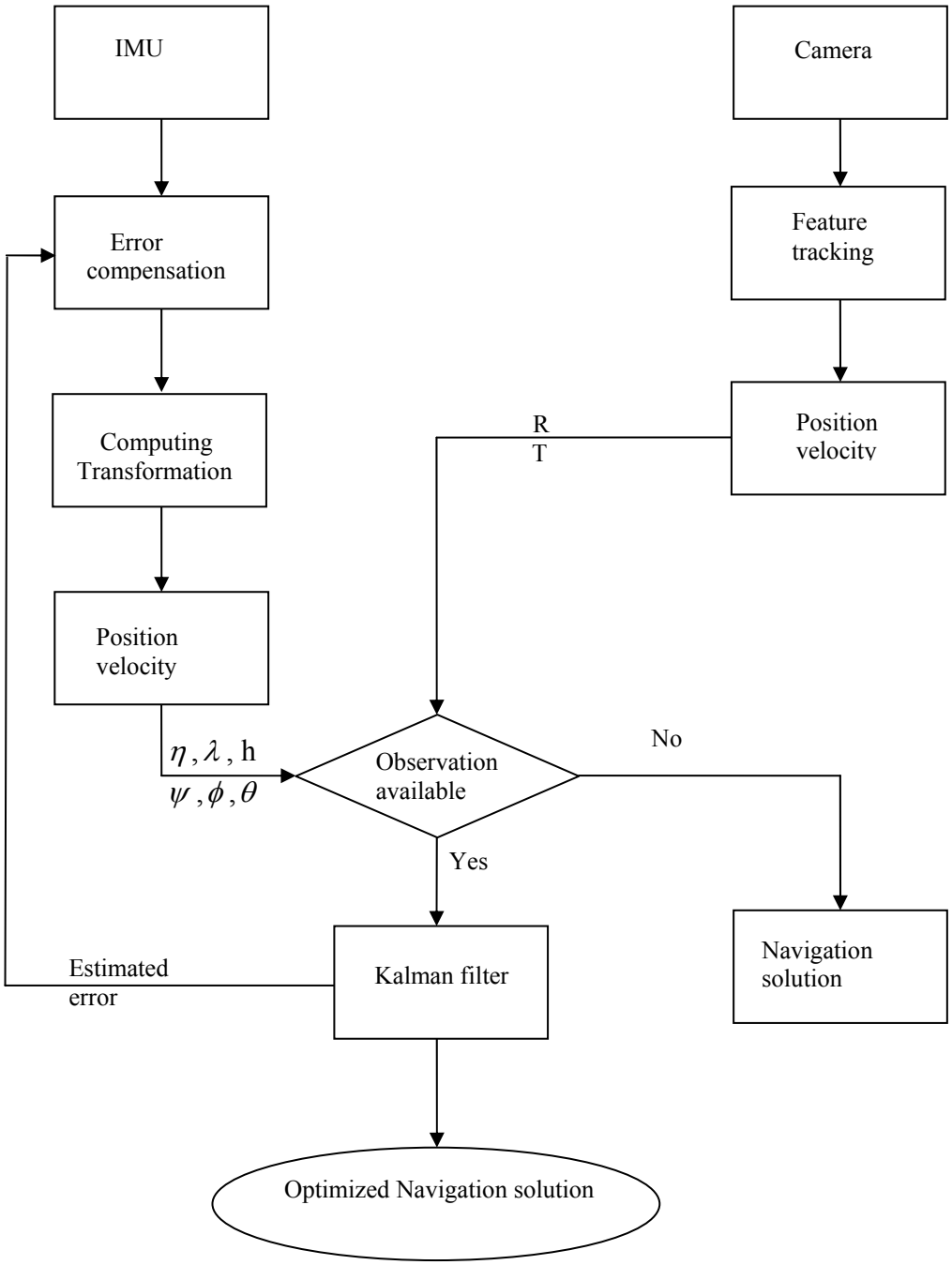


Figure 27: Illustration of processes involved in the fusion algorithm.

Here (Fig. 27) the position and velocity obtained from the vision algorithm is fused with the position and orientation measured from the IMU. In other words, the inputs to the Kalman filter are position and orientations obtained from two different sensor systems. Therefore, this filter has a close resemblance to the decentralized Kalman filter described in (Wei 1990) if merely the format of the input is considered. This form of fusion, using a decentralized filter, is further supported by the local filter designed for the vision sensor system. The local vision filter, or the vision only Kalman filter, plays a prominent role in the overall fusion process due to the high measurement noise in the vision system as illustrated in Fig. 28. In Fig. 28 the translations obtained from the IMU/GPS system are compared with those obtained from the vision system are compared. The high variability in the vision system corresponds to the high noise associated with the vision estimation.

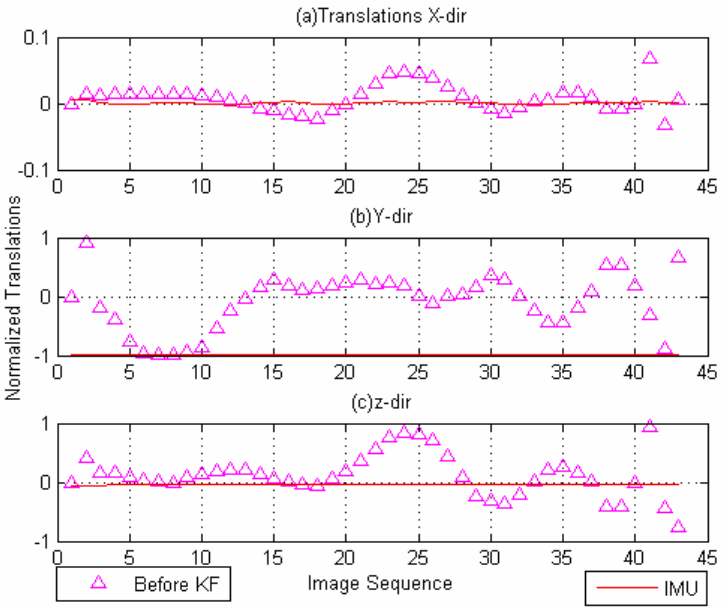


Figure 28: Variability of vision translations due to high noise.

Therefore, the fusion algorithm used in fusing the IMU and the vision systems can be illustrated in Fig. 29;

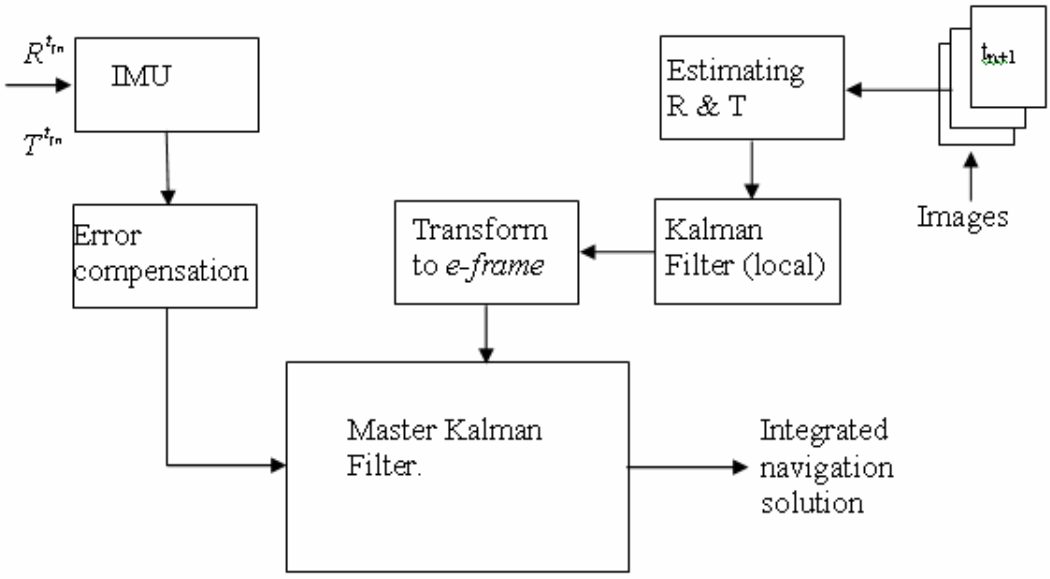


Figure 29: The final fusion architecture of the IMU/Vision system.

In the following subsections a general description of the standard Kalman filter is given followed by illustrations of both the vision only Kalman filter and the master Kalman filter designed in this work.

5.2 Standard Kalman Filter Equations

The standard Kalman filter (Kalman 1960) was developed for linear dynamic systems which would give the existing statistically optimized solution available for that system. Due to the versatility of the Kalman filter in almost all the scientific research areas it is used in non-linear systems as well. Kalman filters have been developed to facilitate prediction, filtering and smoothening. However, in this work Kalman filter will be

predominantly used as a filter and a smoother. The typical Kalman filter equations (Grewal 2001, Jazwinski 1970) can be given as,

Updating:

$$x_k = x_k^- + K_k(y_k - H_k x_k^-) \tag{5-1}$$

$$K_k = P_k^- H_k^T (H_k P_k^- H_k^T - R_k)^{-1} \tag{5-2}$$

$$P_k = (I - K_k H_k) P_k^- \tag{5-3}$$

Prediction:

$$x_{(k+1)}^- = \phi_k x_k \tag{5-4}$$

$$P_{(k+1)}^- = \phi_k P_k \phi_k^T + Q_k \tag{5-5}$$

For the state equations,

$$x_k = \phi_k x_{k-1} + u_k \tag{5-6}$$

$$y_k = H_k y_k + v_k \tag{5-7}$$

where, x_k is the state matrix, y_k is the measurement at kth time step, P_k is the error covariance matrix at kth time step, K_k is the Kalman gain at time k, R_k and Q_k is the variances associated to measurement and process noises respectively and subscript (-) denotes the priori estimate. ϕ_k is the state transition matrix while H_k is the measurement sensitivity matrix.

$$u_k \sim N(0, Q_k)$$
$$v_k \sim N(0, R_k) \tag{5-8}$$

For the proposed model H_k and ϕ_k could be non-linear. Therefore, one has to linearize them at the given time.

5.3 Design of Vision Only Kalman (Local) Filter

The *pose* estimated from the vision sensor system is corrupted due to the various noise types present in the *pose* estimation algorithm. Thus, it is important to minimize this noise and optimize the estimated *pose* from the vision system. The vision sensor predictions can be optimized using a local Kalman filter. Kalman filters have been developed to facilitate prediction, filtering and smoothening. In this context it is only used for smoothen out the rotations and translations predicted by the vision algorithm. A brief description of this local Kalman filter designed for the vision system is outlined in this section and a more thorough description can be found in (Grewal 2001, Jazwinski 1970).

The states relevant to this work consist of translations, rates of translations and orientations. Due to the relative ease of formulating differential equations, associated linearity and the ability to avoid 'Gimble-lock', the orientations are expressed in quaternions. Thus, the state vector can be given as;

$$X_k = \left[T_k, \dot{T}_k, q_k\right]^T \tag{5-9}$$

Where, T_k is the translation, q_k is the orientation given in quaternions and $\dot{T}_k$ is the rate of translation, at time k. Then the updating differential equations for translations and quaternions can be given as;

$$\begin{aligned} T_{k+1} &= T_k + \int_{t_k}^{t_{k+1}} \dot{T}_k dt \\ \dot{q}_{k+1} &= \left(\frac{1}{2}\right) \mathsf{A} q_k \end{aligned} \tag{5-10}$$

Where, ***A*** is given in Eqn (2-18). Then the state transition matrix can be obtained as;

$$\boldsymbol{\varphi}_k = \begin{pmatrix} \mathbf{I}_{3x3} & \delta t \mathbf{I}_{3x3} & \mathbf{0}_{3x4} \\ \mathbf{0}_{3x3} & \mathbf{I}_{3x3} & \mathbf{0}_{3x4} \\ \mathbf{0}_{4x3} & \mathbf{0}_{4x3} & \mathbf{A}_{4x4} \end{pmatrix} \tag{5-11}$$

Where, **I** and **0** are the identity and null matrices of the shown dimensions respectively and δt represents the time difference between two consecutive images. The measurements in the Kalman formulation can be considered as the translations and rotations estimated by the vision algorithm. Therefore, the measurement vector can be expressed as,

$$Y_k = \left[T_k, q_k\right]^T \tag{5-12}$$

Hence, the measurement transition matrix will take the form;

$$\mathbf{H}_k = \begin{pmatrix} \mathbf{I}_{3x3} & \mathbf{0}_{3x3} & \mathbf{0}_{4x4} \\ \mathbf{0}_{3x3} & \mathbf{0}_{3x3} & \mathbf{I}_{4x4} \end{pmatrix} \tag{5-13}$$

Once the necessary matrices are setup using Eqns (5-9)-(5-13) and the initial state vector and the initial covariance matrix are obtained, the vision outputs can be smoothed using the Kalman filter equations. The Initial conditions can be defined conveniently based on the IMU output at the starting location of the test section.

5.4 Design of Master Kalman Filter

The Kalman filter designed to fuse the IMU readings and vision measurements continuously evaluates the error between the two sensor systems and statistically optimizes it. A decentralized Kalman filter architecture (Wei 1990, Allerton 2004, Allerton 2005) is used in this work to fuse the two sensor systems. Since the main aim of the integration of the two systems is to correct the high frequency IMU readings for their error growth, the vision system is used as the updated or precision measurement. Hence, the IMU system is the *process* of the Kalman filter algorithm. Since the two sensor systems have two different data gathering frequencies, multi-rate fusion approach (Armesto 2004) has been used in fusing the IMU and the vision systems. On the other hand, the function of the *Vision Only Kalman filter* is to remove the significantly high noise associated with the vision reading because of the relatively high accuracy of measurements demanded by the fusion algorithm. The system architecture of this master Kalman filter is shown in Fig. 30.

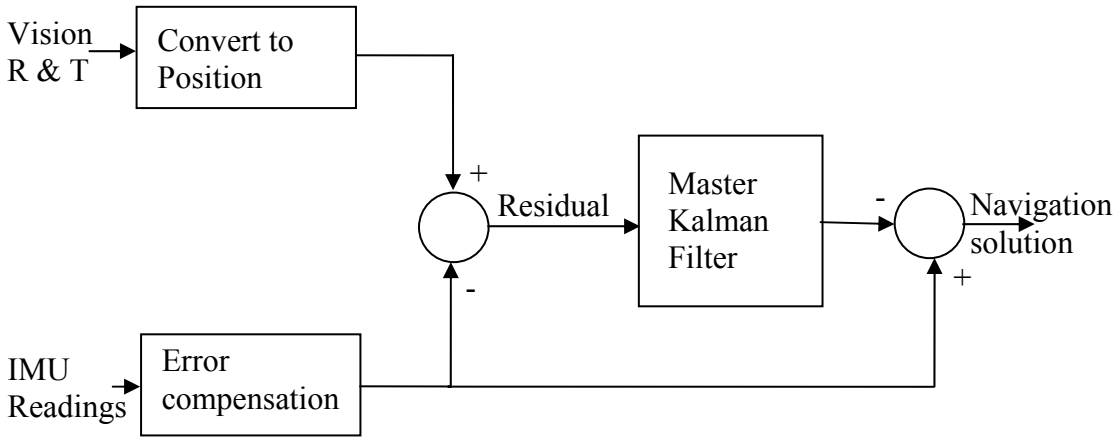


Figure 30: Illustration of master Kalman filter.

The typical inputs to update the master Kalman filter consists of positions (in the *e-frame*) and the orientations of the *b-frame* and the *c-frame* with respect to the *n-frame*. Since the vision system provides rotations and translations between the camera frames, one needs the position and orientation of the first camera location. These can be conveniently considered as respectively the IMU position in the *e-frame*, and the orientation between the *b-frame* and the *n-frame*. The orientation update of the camera between two consecutive images (at t_{k-1} and t_k) with respect to the *n-frame* can be given as;

$$VC_c^n(t_k) = VC_c^n(t_{k-1})R(t_k) \tag{5-14}$$

Where, $VC_c^n(t_k)$ is the transformation matrix between the camera orientation with respect to the *n-frame* at t_k and $R(t_k)$ is the rotation estimated by the Eight Point algorithm at t_k (Faugeras 1996). The IMU used in the test vehicle is a navigational grade IMU which was assumed to be calibrated and aligned quite accurately. Therefore, the main error that

could occur in the IMU measurements is due to biases of gyroscopes and accelerometers. A more detailed explanation of inertial system errors can be found in (Grewal 2001 b). In this work only bias errors were considered in error compensation for the IMU. For gyroscopes and accelerometers, the manufacturer specified bias terms were used in correcting the IMU measurements. These bias terms were considered to be propagating in time as;

$$b_i(t_{k+1}) = b_i(t_k) + w(t_k) \tag{5-15}$$

Where, $b_i(t_k)$ denotes the bias of the ith sensor (i = accelerometer or gyroscope) at time t_k and $w(t_k)$ is a random number. The processing system of the Kalman filter consists of the error terms obtained by perturbation analysis described in Chapter 2. Since these errors are linear, the standard Kalman filter equations can be utilized without any linearization. There are sixteen *(16)* system states used for the Kalman filter employed in the IMU/vision integration. These are; (i) three states for the position, (ii) three states for the velocity, (iii) four states for the orientation, which are given in quaternions and (iv) six states for accelerometer and gyroscope biases. Therefore, the entire state vector for the system (in quaternions) takes the following form;

$$\begin{aligned} X_k = [\delta\phi \quad \delta\lambda \quad \delta h \quad \delta v_n \quad \delta v_e \quad \delta v_d \quad q_w \quad q_x \quad q_y \quad q_z \ldots \\ b_{ax} \quad b_{ay} \quad b_{az} \quad b_{gx} \quad b_{gy} \quad b_{gz}]^T \end{aligned} \tag{5-16}$$

Where, δ denotes the estimated error in the state and v_N, v_E, v_D are respectively the velocity components along the *n-frame* directions while ϕ, λ and h are the latitude, the

longitude and the altitude respectively. The error in the orientation is converted to the quaternion form and its elements are represented as q_i where i= w, x, y, z. And the bias terms in both accelerometers and gyroscopes, i.e. $i=a, b$, along three directions, $j=x, y$ and z, are given as b_{ij}. The system equations in the form of Eqns (5-6) and (5-7) are used in the Kalman filter process since the measurements from both the IMU and the vision system are discrete.

The state transition matrix for this problem would be a 16x16 matrix with the terms obtained from the navigation equations given in Eqns. (2-32)-(2-34). The measurements equation is obtained similarly by considering the measurement residual.

$$\mathbf{y_k} = \left[(\mathbf{P_{Vis}} - \mathbf{P_{imu}}) \quad (\mathbf{\Psi_{Vis}} - \mathbf{\Psi_{imu}})\right]^T \tag{5-17}$$

Where, $\mathbf{P}_i$ and $\mathbf{\Psi}_i$ represent the position vector (3x1) given in geodetic coordinates and the orientation quaternion (4x1) respectively measured using the ith sensor system with i = vision or IMU. Then the measurement sensitivity matrix would take the form;

$$\mathbf{H_k} = \begin{bmatrix} \mathbf{I_{3x3}} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I_{4x4}} & \mathbf{0} & \mathbf{0} \end{bmatrix} \tag{5-18}$$

The last critical step in the design of Kalman filter is to evaluate the process (R_k) and measurement (Q_k) variances of the system. These parameters are quite important in that these define the reliability of the Kalman filter on the system and the measurements (Jazwinski 1970). The optimum values for these parameters must be estimated based on the accuracy of the navigation solution or otherwise a noisy input will dominate the filter

output making it erroneous. In this work, to estimate R_k and Q_k, the authors used a separate data set; one of the three trial runs on the same section that was not used in the subsequent computations. The same Kalman filter was used as a smoother for this purpose. This was important specifically for the vision measurements since it involves more noise in its measurements.

When setting up the differential equations to obtain the navigation solution from the Kalman filter, especially the orientation, quaternions are used in order to avoid undesirable 'Gimble-lock' phenomenon associated with the use of Euler angles. In addition, when quaternions are used those can be used to interpolate and obtain any desired orientation in between the known orientations.

CHAPTER 6 – MATHEMATICAL FORMULATIONS FOR SYSTEM CALIBRATION

6.1 Introduction

This chapter is devoted to formulate two main mathematical relationships that will be used in both calibration and fusion algorithms. The importance of these two mathematical formulations is quite evident as the estimations from both the IMU and the vision system need further processing in transforming the measured sensor data into relevant information. In addition these transformations are needed for the calibration and fusion algorithms. The two transformations explained are;

1) Determine the unique mathematical transformation between the inertial and vision sensor measurements.
2) Determination of position residual from the two sensor systems

6.2 Determination of the Unique Mathematical Transformation between the Inertial and Vision Sensor Measurements

Fig. 31 illustrates the procedure to determine the final transformation between the inertial and vision sensors. The details of obtaining the transformation are described below.

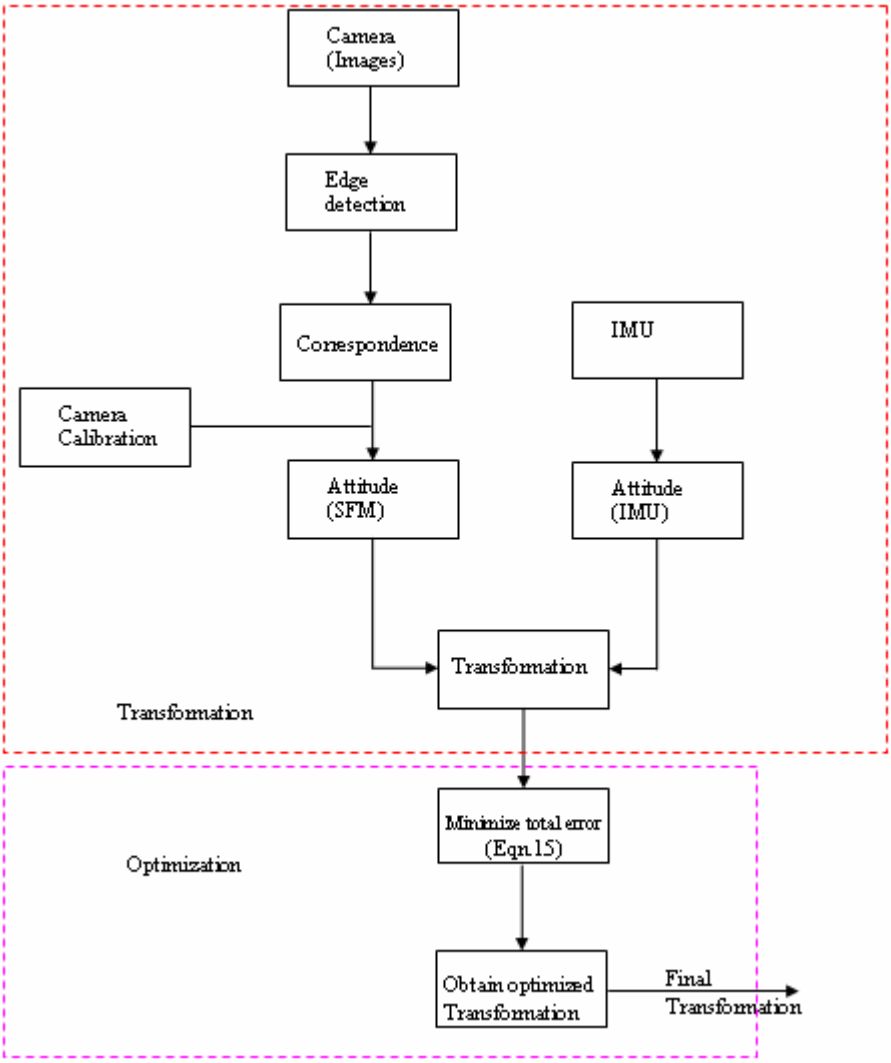


Figure 31: Steps involved in estimating the transformation between the two sensor systems.

The calibration process involves estimating the *pose* of the systems, inertial and vision, at designated locations. These locations can be marked on the pavement (Fig. 32) by special reflective tapes placed at known distance intervals. The FDOT survey vehicle is equipped with a triggering system that is activated to record an event on all the measurement systems when the latter encounters these tapes. Therefore, the *pose* measured by the IMU at the exact tape location can be captured. Also, since the markings are identified on the

pavement, it is possible to evaluate the *pose* from a manual survey at these locations which would be helpful in the verification of the accuracy of the transformation.

Figure 32: Measurement points marked on the pavement by reflective tapes.

However, capture of images cannot be triggered by the above events (reflective tapes) and hence invariably there will be a time offset between the event positions and the image recording positions as illustrated in Fig. 33. This problem can be addressed by interpolating the *pose* estimation obtained by the images. Since it is more accurate to interpolate *pose* estimations in the quaternion space (Shoemake 1985), spherical linear interpolation (SLERP) is used to predict the *pose* of the marked locations with respect to the camera. This is possible since the images also depict the location of tapes even though the camera cannot be triggered by them. The quaternion at any intermediate point can be expressed by the following (SLERP equation) as;

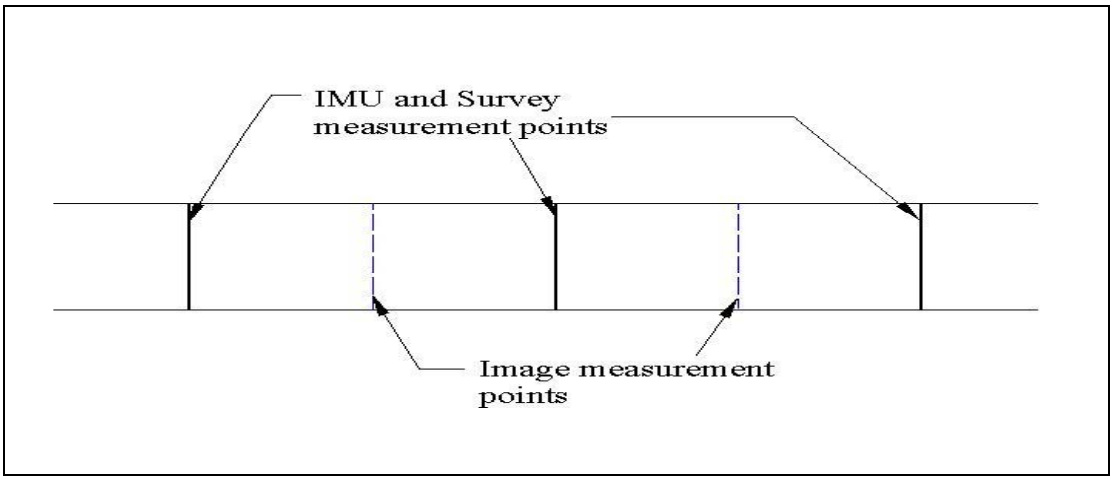


Figure 33: Illustration of the coincidence of IMU and survey measurement points and the offset of image measurements.

$$slerp(q_1, q_2, t) = \frac{\sin((1-t)\theta)}{\sin(\theta)} q_1 + \frac{\sin(t\theta)}{\sin(\theta)} q_2 \qquad (6\text{-}1)$$

$$\cos(\theta) = q_1 . q_2$$

Where, q_1 and q_2 are the known quaternions at the extremes, θ is the angle between the two quaternions and t is any scalar parameter such as time that relates the extremes and intermediate points.

6.2.1 Determination of the Vision-Inertial Transformation

The unique transformation between the two sensor coordinate frames can be determined using a simple optimization technique. In this work it is assumed that the two frames have the same origin but different orientations. First, the orientation of the vehicle at a given position measured with respect to the inertial and vision systems is estimated. Then an initial transformation can be obtained from these measurements. At the subsequent measurement locations, this transformation is optimized by minimizing the total error

between the transformed vision data and the measured inertial data. The optimization produces the unique transformation between the two sensors.

In extending the calibration procedures reported in (Horn 1987, Alves 2003, Lang 2005), modifications must be made to the calibration equations in (Horn 1987, Lang 2005) to incorporate the orientation measurements, i.e. roll, pitch, and yaw, instead of 3D position coordinates. The transformations between each pair of the right-handed coordinate frames considered are illustrated in Fig. 34. In addition, the time-dependent transformations of each system relating the first and second time steps are also illustrated in Fig. 34. It is shown below how the orientation transformation between the inertial and vision sensors ($\mathbf{R}_{vi}$) can be determined by using measurements which can easily be obtained at an outdoor setting.

In Fig. 34 OG, OI, and OV denote origins of global, inertial, and vision coordinate frames respectively. xk, yk, and zk define the corresponding right handed 3D-axis system with k representing the respective coordinate frames (i-inertial, v-vision, and g-global). Furthermore, the transformations from the global frame to the inertial frame, global frame to the vision frame and inertial frame to the vision frame are defined respectively as $\mathbf{R}_{ig}$, $\mathbf{R}_{vg}$, and $\mathbf{R}_{vi}$.

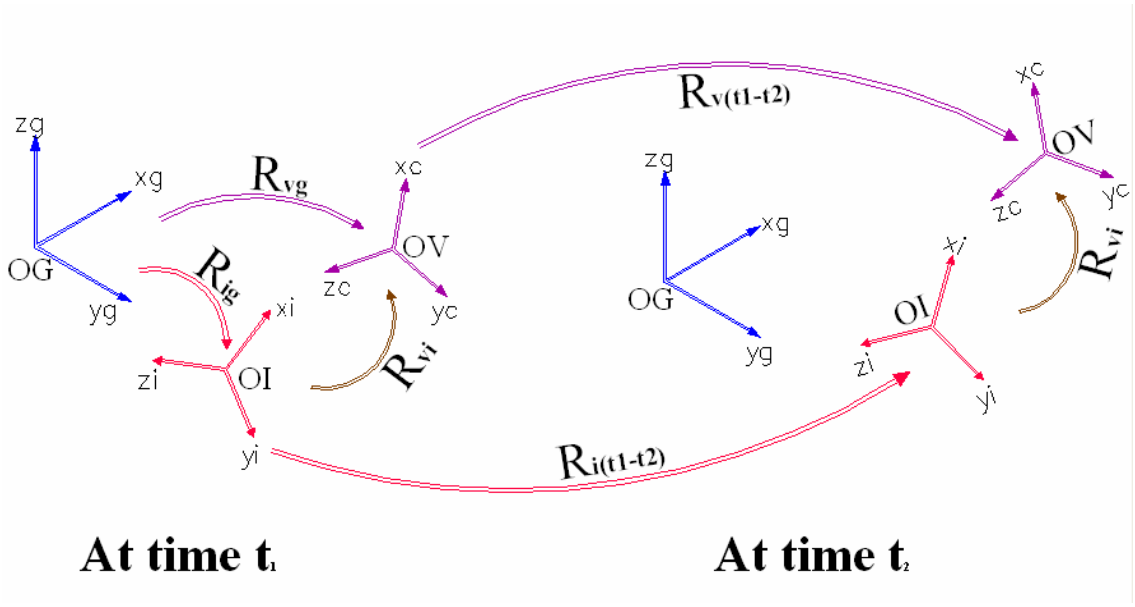


Figure 34: Three coordinate systems associated with the alignment procedure and the respective transformations.

If $\mathbf{P}_g$ denotes the position vector measured in the global coordinate frame, the following equations can be written considering the respective transformations between the global frame and both the inertial and the vision frames.

$$\mathbf{P}_{g(t1)} = \mathbf{R}_{ig(t1)}\mathbf{P}_{i(t1)} \tag{6-2}$$

$$\mathbf{P}_{g(t1)} = \mathbf{R}_{vg(t1)}\mathbf{P}_{v(t1)} \tag{6-3}$$

And considering the transformation between the inertial (OI) and vision systems (OV);

$$\mathbf{P}_{i(t1)} = \mathbf{R}_{vi}\mathbf{P}_{v(t1)} \tag{6-4}$$

Substituting Eqn (7-2) and Eqn (7-3) into Eqn (7-4), the required transformation can be obtained as;

$$\mathbf{R}_{vi} = \mathbf{R}_{ig(t1)}^{-1}\mathbf{R}_{vg(t1)} \tag{6-5}$$

Although the transformations between global-inertial and global-vision are time variant, the transformation between the inertial system and the vision system ($\mathbf{R}_{vi}$) is time invariant due to the fact that the vision and inertial systems are rigidly fixed to the vehicle. Once the *pose* estimates for IMU and vision are obtained, the corresponding rotation matrices (in the Euler form) can be formulated considering the rotation sequence of 'zyx'. Thus, Eqn (6-5) provides a simple method of determining the required transformation $\mathbf{R}_{vi}$. Then the Euler angles obtained from this step can be used in the optimization algorithm as initial angle estimates. These estimates can then be optimized as illustrated in the ensuing section to obtain more accurate orientations between x, y, and z axes of the two sensor coordinate frames.

6.2.2 Optimization of the Vision-Inertial Transformation

If α, β, and γ are the respective orientation differences between the axes of the inertial sensor frame and the vision sensor frame, then the transformation $\mathbf{R}_{vi}$ can be explicitly represented in the Euler form by $\mathbf{R}_{vi}(\alpha,\beta,\gamma)$. Using Eqn (6-5) the rotation matrix for the inertial system at anytime t' can be expressed as;

$$\mathbf{R}^{*}_{ig(t')} = \mathbf{R}_{vg(t')}\mathbf{R}^{-1}_{vi}(\alpha,\beta,\gamma) \tag{6-6}$$

$\mathbf{R}_{vg(t')}$ can be determined from a sequence of images obtained using the algorithm provided in (Taylor 1995, Faugeras 1996) and $\mathbf{R}^{*}_{ig(t')}$ can be estimated using Eqn (6-6) for any given set (α,β,γ). On the other hand, one can determine $\mathbf{R}_{ig(t')}$ directly from the IMU measurements. Then a non-linear error function (e) can be formulated in the form;

$$e_{pq}^2(\alpha,\beta,\gamma) = [(\mathbf{R}_{ig(t')})_{pq} - (\mathbf{R}^*_{ig(t')})_{pq}]^2 \tag{6-7}$$

Where p (=1, 2, 3) and q (=1, 2, 3) are the row and column indices of the $\mathbf{R}_{ig}$ matrix respectively. Therefore, the sum of errors can be obtained as;

$$E = \sum_p \sum_q e_{pq}^2(\alpha,\beta,\gamma) \tag{6-8}$$

Finally, the optimum α , β , and γ can be estimated by minimizing Eqn (6-8);

$$\min_{\alpha,\beta,\gamma}\{E\} = \min_{\alpha,\beta,\gamma}\left\{\sum_p \sum_q [(\mathbf{R}_{ig})_{pq} - (\mathbf{R}_{ig(t')})_{pq}]^2\right\} \tag{6-9}$$

Minimization can be achieved by gradient descent Eqn (6-10) as follows;

$$\mathbf{x}_i = \mathbf{x}_{i-1} - \lambda E'(\mathbf{x}_{i-1}) \tag{6-10}$$

Where, $\mathbf{x}_i$ and $\mathbf{x}_{i-1}$ are two consecutive set of orientations respectively while λ is the step length and $E'(\mathbf{x}_{i-1})$ is the first derivative of E evaluated at $\mathbf{x}_{i-1}$;

$$E'(\mathbf{x}_{i-1}) = \left[\frac{\partial E(\mathbf{x}_{i-1})}{\partial \alpha} \quad \frac{\partial E(\mathbf{x}_{i-1})}{\partial \beta} \quad \frac{\partial E(\mathbf{x}_{i-1})}{\partial \gamma}\right]^T \tag{6-11}$$

Once the set of angles (α,β,γ) corresponding to the minimum E in Eqn (6-8) is obtained, for time step t', the above procedure can be repeated for a number of time steps t'', t''' etc. When it is verified that the set (α,β,γ) is invariant with time it can be used in building the unique transformation ($\mathbf{R}_{vi}$) matrix between the two sensor systems.

6.2.3 Verification with the Survey Measurements

Once the unique transformation between the two sensor systems is derived, it can be verified by comparing the predicted vehicle rotation maneuvers with those measured from a manual survey. Orientation of a series of survey locations along the roadway can be estimated with respect to the first survey point. Since the orientations are measured with respect to the first point, a separate formulation is needed to estimate the transformation between the survey-inertial systems. This also can be done based on Fig. 34 by replacing the global frame with the survey frame.

If the transformation between two time steps in any of the frames (Fig. 34) is given as $\mathbf{R}_{j(t1-t2)}$ (j=i-inertial, j=v-vision, and j=s-survey), then the position vector at the second step can be obtained as;

$$\mathbf{P}_{s(t1)} = \mathbf{R}_{s(t1-t2)}\mathbf{P}_{s(t2)} \tag{6-12}$$

The transformations in the vision and inertial frames between the first and second time steps can be also expressed in a similar fashion as:

$$\mathbf{P}_{i(t1)} = \mathbf{R}_{i(t1-t2)}\mathbf{P}_{i(t2)} \tag{6-13}$$

By using Eqn. (6-5);

$$\mathbf{P}_{s(t2)} = \mathbf{R}_{is(t2)}\mathbf{P}_{i(t2)} \tag{6-14}$$

Eqns (6-2), (6-12) and (6-14) can be combined to eliminate $\mathbf{P}_{i(t1)}$, $\mathbf{P}_{i(t2)}$, $\mathbf{P}_{s(t1)}$, and $\mathbf{P}_{s(t2)}$ to obtain the survey-inertial frame for the second time step as:

$$\mathbf{R}_{is(t2)} = \mathbf{R}^{-1}_{s(t1-t2)}\mathbf{R}_{is(t1)}\mathbf{R}_{i(t1-t2)} \tag{6-15}$$

Similarly, the vision-survey transformation can be deduced as;

$$\mathbf{R}_{vs(t2)} = \mathbf{R}^{-1}_{s(t1-t2)}\mathbf{R}_{is(t1)}\mathbf{R}_{i(t1-t2)}\mathbf{R}_{v(t1\text{-}t2)}\mathbf{R}^{-1}_{vi}(\alpha,\beta,\gamma) \tag{6-16}$$

Where, $\mathbf{R}_{vs(t2)}$ is the vision measurement transformed onto the survey frame at the time step t_2. While $\mathbf{R}_{i(t1\text{-}t2)}$ and $\mathbf{R}_{v(t1\text{-}t2)}$ can be obtained from vehicle kinematics, $\mathbf{R}_{s(t1\text{-}t2)}$ can be established based on surveying. Since the vision measurements can be transformed to the inertial frame using Eqn (6-5) and then to the survey frame by using Eqn (6-16), they can be compared with survey measurements to determine the accuracy of the estimated transformation.

6.3 Determination of Position Residual from the Two Sensor Systems

Although one can obtain translations between the camera positions from a sequence of images collected using the vision sensor, the translations estimations are scaled in the z direction of the *c-frame* (Fig. 12b). In other words, due to the perspective projection of the 3D points onto the image plane, the total recovery of the depth is not allowed. Therefore, the translations derived from the vision algorithm are indeed the normalized translations or unit vectors providing only the directional information and not the actual magnitude.

One of the inputs (measurements) to the Kalman filter that executes the fusion between the IMU and vision information is the position residual estimated by those two sensors. Of these, the IMU provides a vector expressing the actual translations while the vision

sensor expresses the translation as a unit vector. Thus the fusion of vision and inertial measurements requires this vision-based translation to be expressed as a regular vector rather than a unit vector. Hence, a special technique had to be devised to obtain the position residual. The method followed in this work to estimate the measurement residual is explained below.

In this work, the authors first transform both measurements (IMU and vision) onto the *e-frame* (Fig. 11). Then, the translations measured by both sensors are projected onto a unit sphere (Fig. 35) and the two respective unit vectors and hence the difference between them that would produce the measurement residuals can be estimated. Finally, as needed by the input to the fusion algorithm, these unit residuals are multiplied by the magnitude of the IMU based translation vector.

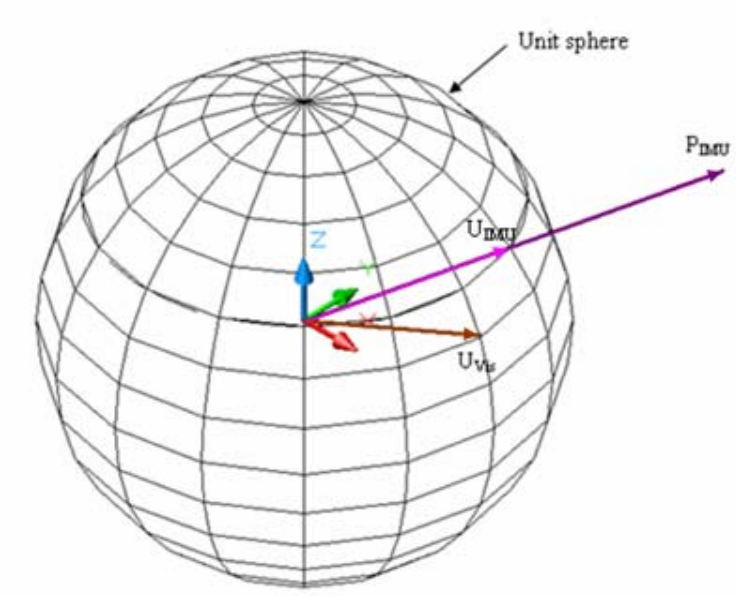


Figure 35: Illustration of measurement residual estimation.

In Fig. 35, P_{IMU} is the position vector between two consecutive camera locations in the *e-frame* as estimated by the IMU whereas U_{vis} and U_{IMU} denote the unit translation vectors estimated from the vision system and the IMU respectively, transformed to the *e-frame*. First, the transformation of the vision system measurements from the *c-frame* (Fig. 12b) into the *e-frame* (Fig.11) can be performed as;

$$U_{Vis}^{e}(t_k) = C_n^e(t_k)C_b^n(t_k)C_c^b T^c \tag{6-17}$$

Where, C_c^b is the transformation between the *c-frame* and the *b-frame*, which can be obtained using the vision system calibration procedure outlined in Chapter 5 and C_b^n can be estimated by Eqns (2-17) – (2-19). The superscript *e* indicates that the quantities are expressed in the *e-frame*. Transformation between the *n-frame* and the *e-frame* can be obtained by considering the instantaneous latitude and the longitude. It can be deduced that,

$$C_n^e = R_3(-\lambda(t_k))R_2(\pi/2 + \phi(t_k)) \tag{6-18}$$

Where, R_i represents the Euler rotation about the i^{th} axis (i =2, 3). The first camera location can be established as the corresponding IMU position in the *e-frame*. Then the position at any other time is estimated using Eqn (6-17). Similarly, the IMU translations can be transformed into the *e-frame* by;

$$P_{IMU}^{e}(t_k) = C_n^e(t_k)C_b^n(t_k)T_{IMU}^b \tag{6-19}$$

Where the term T_{IMU}^{b} is the translation estimated by the IMU measurements between two consecutive camera locations. The IMU position vector obtained from Eqn (6-19) is then normalized and the unit vector, $U_{IMU}^{e}(t_k)$, associated with the IMU is determined (Fig. 35). Once the two unit vectors are estimated the measurement residual, an input to the fusion filter, is obtained by;

$$dT^{e} = \left|P_{Vis}^{e}(t_k)\right|(U_{Vis}^{e}(t_k) - U_{IMU}^{e}(t_k)) \tag{6-20}$$

Where, $|P_{Vis}^{e}(t_k)|$ is the magnitude estimated from the IMU measurement and dT^{e} is the required translations residual in the *e-frame*.

CHAPTER 7 – EXPERIMENTAL SETUP AND RESULTS

7.1 Experimental Setup

The data for the fusion process was collected on a test section on Eastbound State Road 26 in Florida. The total test section was divided into three separate segments; one short run and two other relatively longer runs. On the short segment, data from three measuring frames, i.e. survey, vision, and inertial, were collected. The two longer sections were selected in such a way that they would include the typical geometric conditions encountered on a roadway, such as straight sections, horizontal curves and vertical curves. Data collected at the two longer runs *(1) Straight* and *(2) Horizontal curve* on the State Road 26 was used for the validation purpose. A typical road section used for data collection can be illustrated in Fig. 17. Since the manual survey is relatively expensive and time and labor intensive, it was not performed in the longer segments. The longer segments were demarcated so that there would be an adequate number of sample points in each of them.

7.2 Experimental Verification of Highway Geometric Data

7.2.1 Cross Slope Data

Two experimental data sets obtained by the FDOT highway evaluation vehicle on two separate test locations were used to verify the algorithms used in the INS/DGPS system.

The first data set was obtained during a test run on Interstate-10 in Florida where cross-slope data measured by a manual survey ware also available. The second data set obtained on a reverse curve along the NE 8th Avenue in Gainesville, Florida contained the entire raw roadway geometric data obtained using a PC card. Fig. 36 illustrates the parabolic fit obtained for a section within the former test section. Tables 4 and 5 compare the results obtained in two consecutive runs using the three different methods, (1) slope obtained from the manufacturer's (ICC) software (2) slope obtained by the manual survey (3) slopes obtained by LSA fits, both linear and parabolic, for the same section.

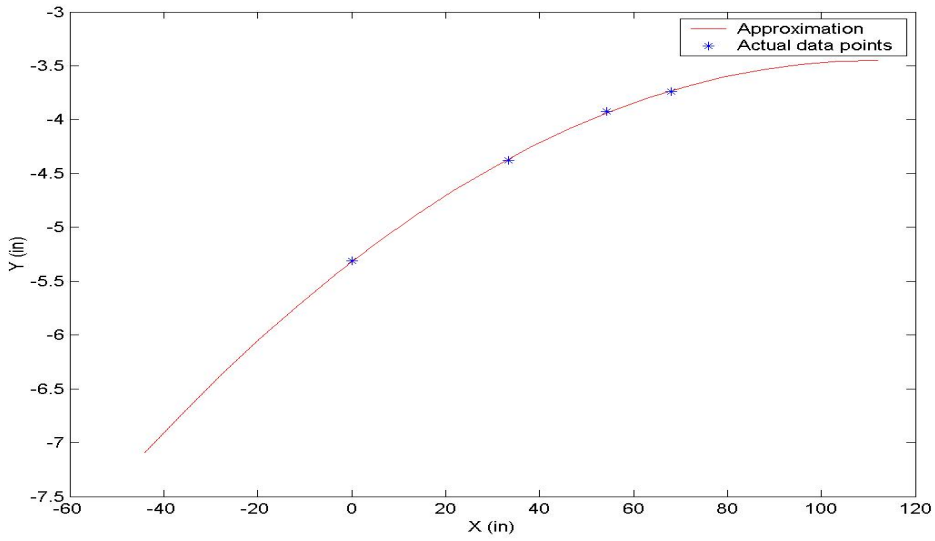


Figure 36: Parabolic fit for a point along the test section, 50 ft away from the starting point of the test section.

Table 4: Comparison of cross-slopes for first run on I-10.

Distance (ft.)	ICC slope (%)	Survey slope (%)	Linear slope (%)	Parabolic slope (%)	Error (%)			Correlation coefficient	
					ICC	Linear fit	Para. fit	Linear	Para.
50	1.123	2.46	2.300	2.273	54.35	6.50	7.60	0.9805	1
200	0.676	1.44	2.010	1.996	53.06	39.60	38.61	0.9914	0.9969
800	-0.996	-1.84	-1.774	-1.792	45.87	3.57	2.61	0.9891	0.9985

Table 5: Comparison of cross-slopes for second run on I-10.

Distance (ft.)	ICC slope (%)	Linear slope (%)	Parabolic slope (%)	Error (%)			Correlation coefficient	
				ICC	Linear fit	Para. fit	Linear fit	Para. Fit
50	1.248	2.362	2.337	49.27	3.98	5.00	0.9847	0.9997
200	0.450	1.049	1.034	68.75	27.15	28.19	0.9641	0.9884
800	-1.006	-1.531	-1.550	45.33	16.80	15.76	0.9859	0.9999

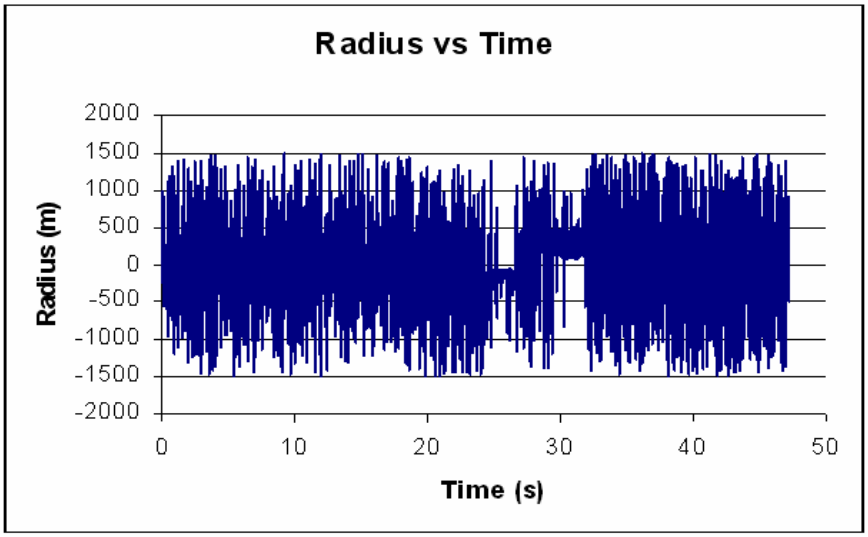
Results in Table 4 and 5 show reasonable repeatability except at the location of 200ft from the start. In both Tables 4 and 5 the International Cybernetics Cooperation (ICC) slope is obtained from the manufacturer's software using Eqn (3-9). The third column of Table 4 provides the slope determined from the manual survey while the fourth and the fifth columns contain the slopes obtained by the linear and parabolic LSA fits, using Eqns (3-11) and (3-12). It is seen that the LSA parabolic fit approximates the roadway cross-slopes much more accurately than any other method. The significant differences observed between the surveyed cross-slope values and all of the fitting methods at a distance of 200 ft from the starting location can possibly be attributed to the inaccurate identification of the manually surveyed location. This hypothesis is further supported by the unsatisfactory repeatability of IMU measurements at this location observed in Table 4 and 5.

7.2.2 Radius of Curvature Data

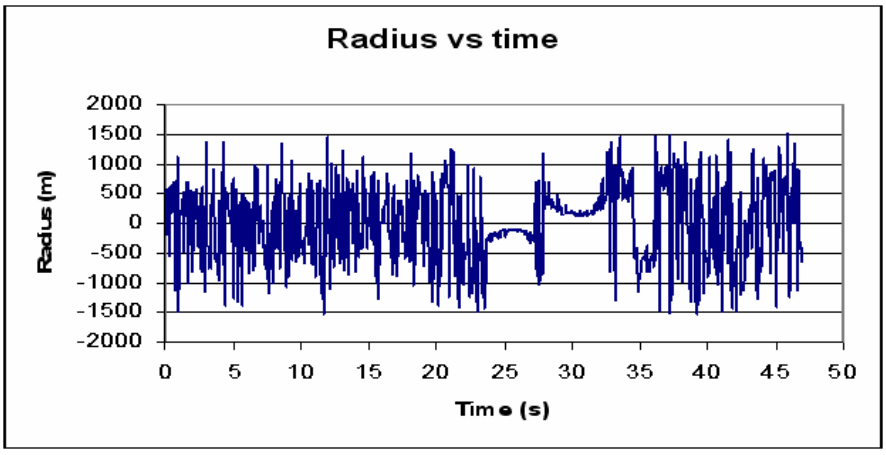
Data obtained from the test section on a reverse curve at NE 8th Avenue in Gainesville, Florida was used to test the algorithms for determining the radius of curvature.

Application of the geometric method of determining the radius

Fig.37(a) provides the variation of radius with time based on the algorithm in Eqn (3-12),



(a)



(b)

Figure 37: Radius calculated by the geometric method for a data acquisition frequency of 200 Hz (a) without averaging, (b) with averaging.

Figure 38 shows the variation of the radius with time calculated based on $v_{\theta,}$ the velocity in the longitudinal direction and $\dot{\gamma}$, the first derivative of platform heading with respect to time (Eqn (3-9b)) determined using Eqn (3-13).

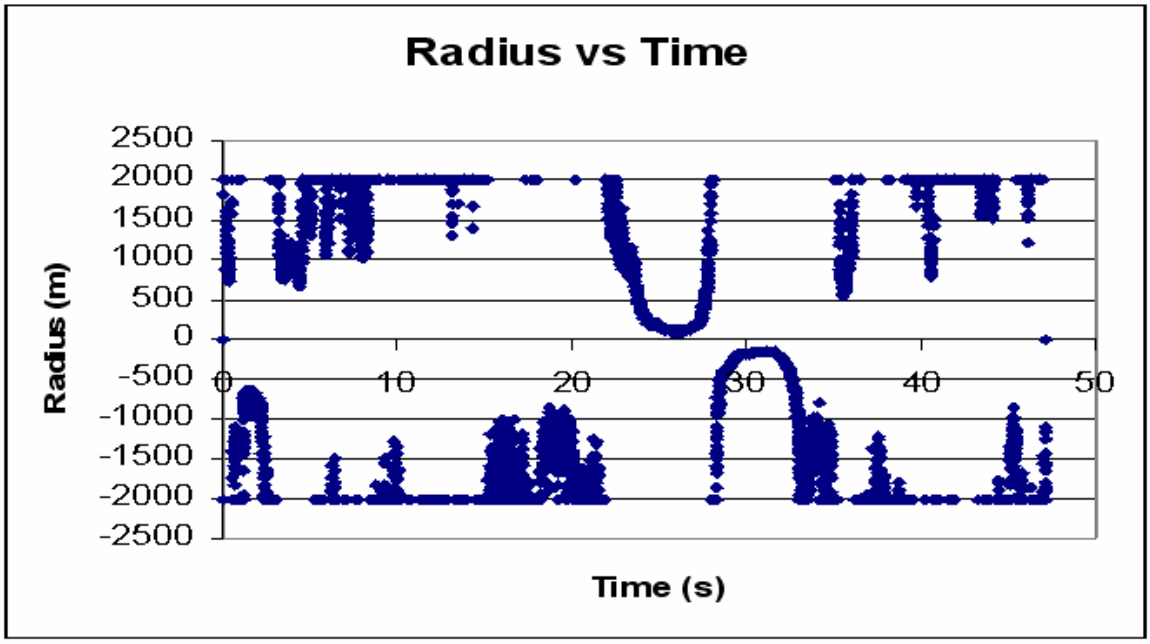


Figure 38: Variation of radius with time evaluated from Eqn (3-12) (frequency 200 Hz).

In Figure 38 a cut-off value of 2000 meters is imposed on the radius to overcome the issues of undulations of large radii of curvature on the tangent sections. In comparing Fig. 38 with Fig 37 (a) and (b), it is seen that the former plots the radius with less variability. Furthermore, both the tangent sections as well as the curved sections show less oscillation and are easily distinguishable.

Application of the kinematic method of determining the radius

Fig.39 shows a typical plot of predicted body acceleration in the lateral (Y) direction (a_{by}) which also exhibits significant undulations especially within both tangents and the curves. In order to separate the tangents from the curves, upper and lower bounds of ($\mu \pm$

3σ) were used. μ and σ are the mean and the standard deviation of the lateral acceleration data obtained from the tangent sections.

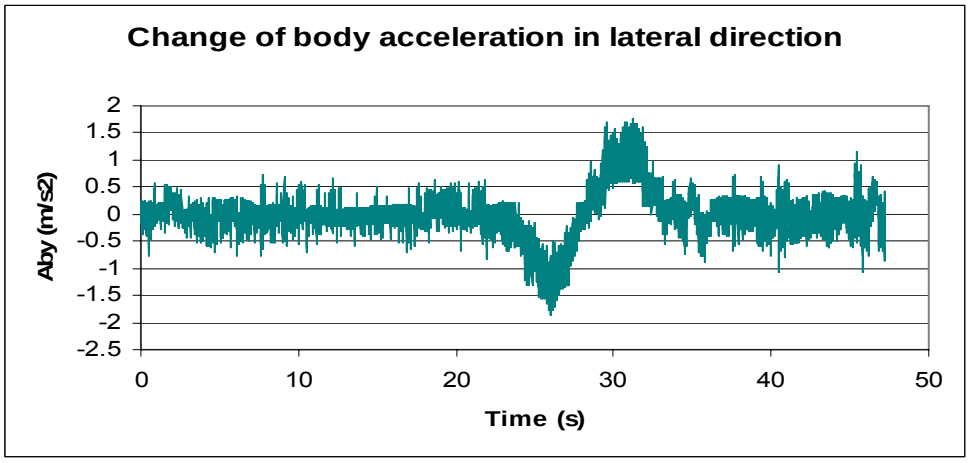


Figure 39: Plot of body acceleration in the lateral (Y) direction with time for the total section.

The acceleration values of the tangent section in Fig. 39 consistently lie within the upper and lower bounds defined for the tangent section. It is seen that this pattern changes once the vehicle enters a curve where the lateral acceleration component remains deviated from these bounds. By applying the above technique to the lateral acceleration record, it is possible to demarcate curved sections from tangent sections. Figure 40 shows radii of curvature computed after separating two curves from the linear sections using Eqn (3-12). From Figure 40 one can also clearly see the change in radius within the transition curves and constant radii in the circular sections.

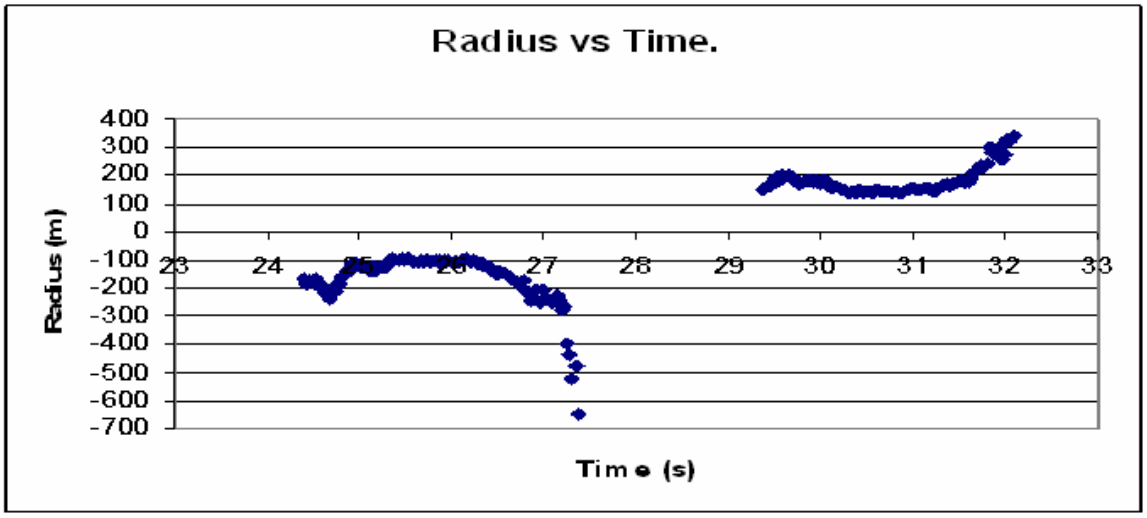


Figure 40: Curved sections separated from the tangent sections, using upper and lower bounds.

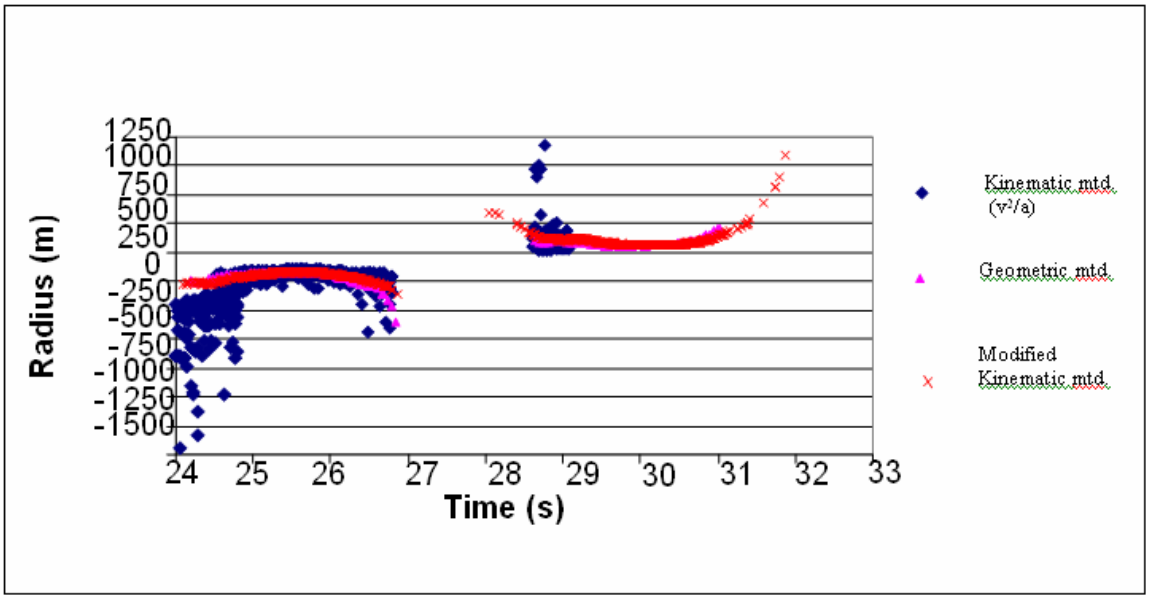


Figure 41: Comparison of radii values calculated by (1) using kinematic method, (2) using geometric method, (3) using modified kinematic method.

Finally, in Fig.41 the radii values estimated by the three different approaches are compared, (1) the kinematic method using Eqn (3-8), (2) the geometric method using Eqn (3-12) and (3) modified kinematic method using Eqn (3-12). It is clear that the radii values match well within the parts of the curve which are circular but deviate in the

transition sections. The results emphasize the fact that the most accurate and stable solution for the transition sections are provided by the modified kinematic method.

Table 6: Comparison of radius values obtained for the two curves.

	Manual method (m)	Compass method (m)	Geometric Method (m)	Kinematic method (m)	Modified kinematic method (m)
Curve 1	108.63	109.30	102.73	113.83	112.50
Curve 2	152.82	151.17	143.08	162.45	157.28

Radii computations from all of the methods are tabulated in Table 6. From Table 6 one can also see that the two curve segments of the reverse curve in Fig. 41 have different radii values. Figure 42 shows the radius of curvature calculated for curve 1 and curve 2 by using the compass method described in Section IV. The angular difference at two consecutive points is obtained from the platform heading and the distance between two points was estimated by using Eqn. (3-11). Once the angular difference and the distance between two points are available the radius is calculated using Eqn (3-18).

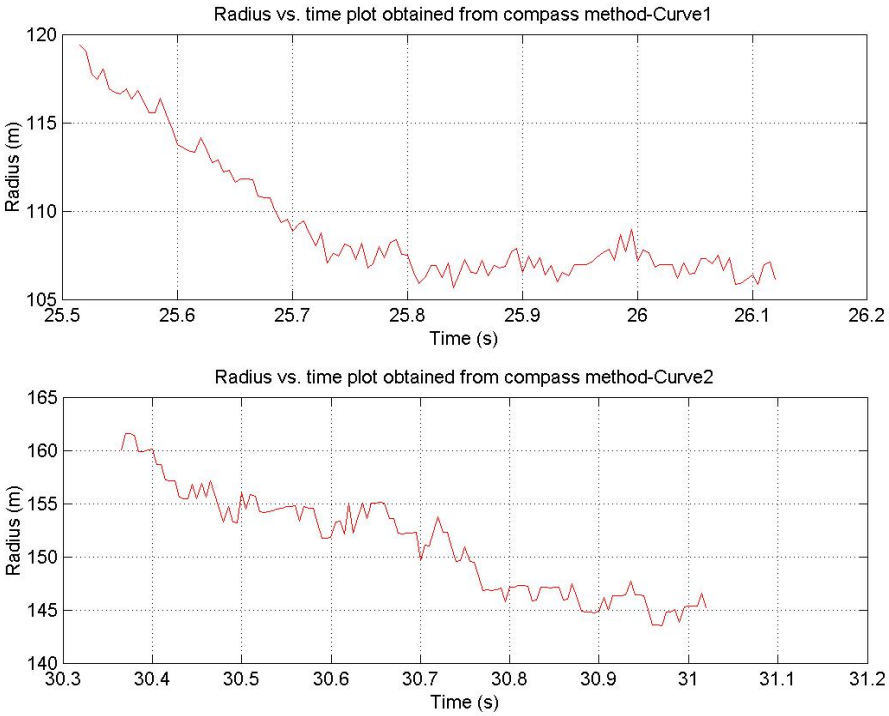
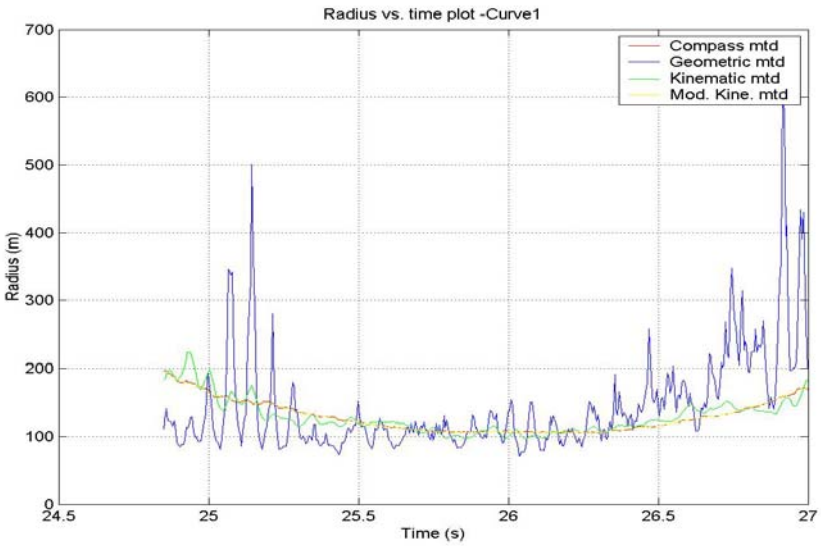
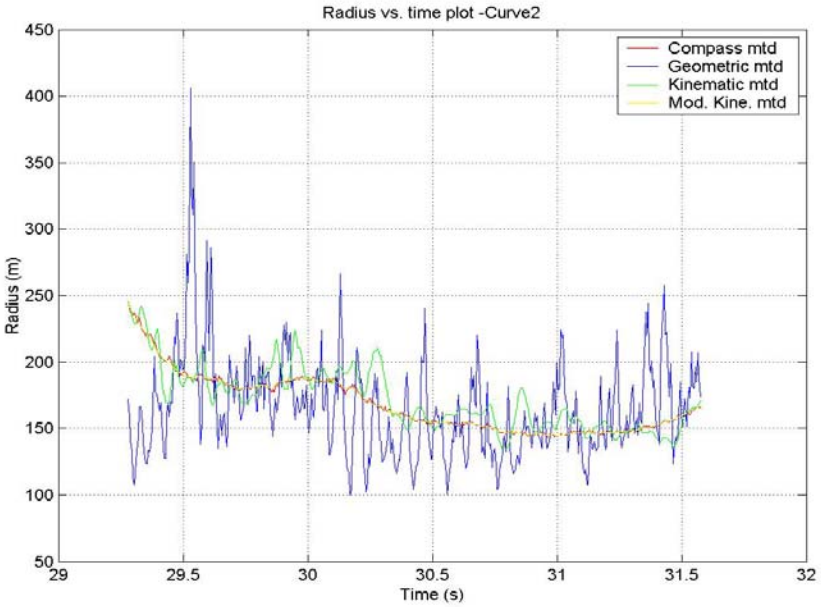


Figure 42: Radius values obtained by compass method for two curves.



(a)



(b)

Figure 43:Comparison of radius values obtained from 1) compass method, 2) geometric method, 3) kinematic method and 4) modified kinematic method for (a) curve 1 and (b) curve 2 including transition curves.

7.3 Calibration for Intrinsic Properties of the Vision System

A 2D calibration target (Fig. 44) consisting of a pattern of circular control points was used to calibrate the camera for its intrinsic properties discussed in Section V. In order to evaluate the above six intrinsic properties an adequate number of circular data points was distributed on the target. These circles were setup to have radii values of 1 cm and a suitable center to center spacing so that it is possible to use the algorithm in (Heikkila 2000) directly to determine the intrinsic parameters of the camera. Once the intrinsic parameters of the camera are estimated they can be used to depict the distortion that occurs in the vision system, by plotting the distorted data against the original data.

The camera of the FDOT survey vehicle has a field of view of 55^{o}-65^{o} and is mounted on the hood of the vehicle, about 7.5 ft above the ground (Fig. 1). Therefore, it is impractical to position the target for the camera to capture the target in its entirety. Thus, the images were captured in such a way that the target would occupy the maximum practically possible area of the image.

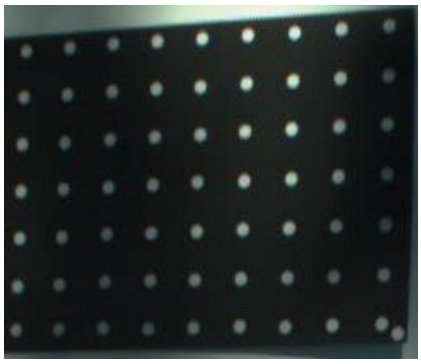


Figure 44: Calibration target with circular control points.

Table 7 summarizes the intrinsic properties for the vision system in the survey vehicle. Table 7 also shows that there is in fact significant distortion occurring in the images in the radial and tangential forms. Also the pixel coordinates of the principal point, scale factor, and focal length in Table I would be useful parameters in the accurate estimation of *pose* from vision.

Table 7: Intrinsic parameters of the vision system.

Intrinsic Property		Value
Focal Length (mm)		24.53
Scale Factor		0.79
Principal Point	X_0 (pixels)	693.23
	Y_0 (pixels)	495.52
Radial Distortion	k_1 (mm^{-2})	-2.19×10^{-2}
	k_2 (mm^{-4})	2.04×10^{-3}
Tangential Distortion	T_1 (mm^{-1})	1.16×10^{-2}
	T_2 (mm^{-1})	2.38×10^{-3}

Fig. 45 shows the image of the calibration target shown in Fig. 44 in which the original input data points (u_d,v_d) are darkened while the output from the distortion model (Heikkila 2000) (u_c,v_c) is shown as circles. Fig. 45 clearly depicts the magnitude of the distortion in both the radial and tangential directions. Based on Fig. 45 it can be concluded that the corrected data points match reasonably well with the corresponding locations on the target.

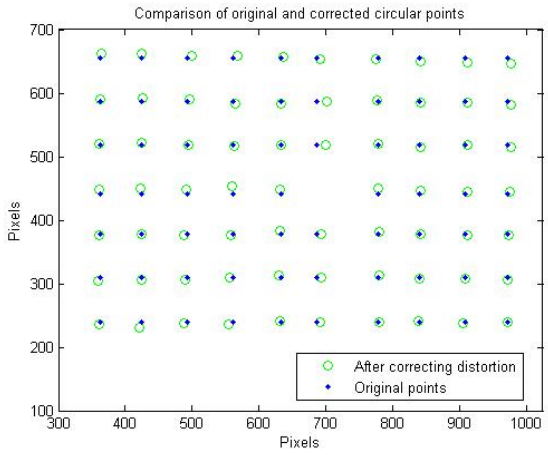


Figure 45: Comparison of original and the distorted data points from the model on the calibration target.

7.3.1 Estimating the Transformation between Inertial Sensor and Vision Sensor

Vision data was collected at all three test segments, one short segment and two relatively longer segments, setting the image capturing distance interval to be 5 ft. This corresponds to the highest allowable data logging frequency of the system under normal operating speeds. Meanwhile, the IMU frequency can be set as high as 200 Hz. Once the images are collected, the vision *pose* from these images can be estimated as discussed in chapter 5. *Pose* from the inertial sensor is available in the desired format in the manufacturer's post-processing software, *Applanix PosPAC*. Due to the different data logging frequencies of the two sensor systems and since the reflective tape cannot activate the vision system at the desired location, the *pose* estimated from the vision system is interpolated to obtain the *pose* at the taped locations (Fig. 32) using Eqn (6-11). The transformation illustrated in Fig. 34 was used to convert vision data, in terms of roll, pitch and yaw, to the inertial frame for comparison with actual inertial data. Since the data logging frequency of the inertial sensor is much higher than that of the vision sensor, the

data from the inertial sensor was extracted at locations where vision data is available, for comparison with the converted vision data.

The inertial and vision data collected on the longer test sections was used for the development and validation of the transformation described in Sections VII*(2)* and *(3)* while data from the short run, i.e. inertial, vision and survey, was used to verify the transformation with the ground-truth.

Table 8 summarizes the optimized transformations obtained for the inertial-vision system. It shows the initial estimates used in the optimization algorithm (Eqn 6-9) and the final optimized estimates obtained from the error minimization process at three separate test locations (corresponding to times t', t'' and t'''). It is clear from Table 8 that the optimization process converges to a unique (α, β, γ) set irrespective of the initial estimates provided. Since the two sensor system is rigidly fixed to the vehicle, the inertial-vision transformation must be unique. Therefore, the average of the optimized transformations can be considered as the unique transformation that exists between the two sensor systems.

Table 8: Orientation difference between two sensor systems estimated at four locations.

		Initial Angle	Optimized Angle
Point 1 (t')	Roll (rad)	-0.00401	-0.03304
	Pitch (rad)	-0.00713	0.01108
	Yaw (rad)	1.23723	-0.08258
Point 2 (t'')	Roll (rad)	-0.03101	-0.03304
	Pitch (rad)	-0.00541	0.01108
	Yaw (rad)	1.34034	-0.08258
Point 3 (t''')	Roll (rad)	-0.01502	-0.03304
	Pitch (rad)	-0.00259	0.01108
	Yaw (rad)	1.32766	-0.08258

7.3.2 Validation of Results

Once the *pose*s from the inertial and vision systems were obtained as mentioned in Chapter 2 and 4, the transformation determined in Chapter 6 was applied to the vision data and the transformed vision data was compared with the inertial sensor data. It must be noted that this transformed output occurs in its raw format whereas the inertial sensor data have already been statistically filtered. Therefore, the inertial data tend to be smoother compared to the transformed measurements of the vision *pose*. To address this disparity, a simple moving median filter was used to smoothen out the transformed data of the vision *pose* as well.

The comparison of the transformed vision and inertial data are illustrated in Fig. 46 and Fig. 47 for the straight run and horizontal curve respectively. It is clear that the original inertial data and the transformed vision measurements have reasonably close agreement. The reasons for the observed minor deviation can be attributed to measurement errors,

noise problems (Alves 2003), various outdoor factors like vibrations and above all, the coarseness of the data computational interval which is as large as 5ft.

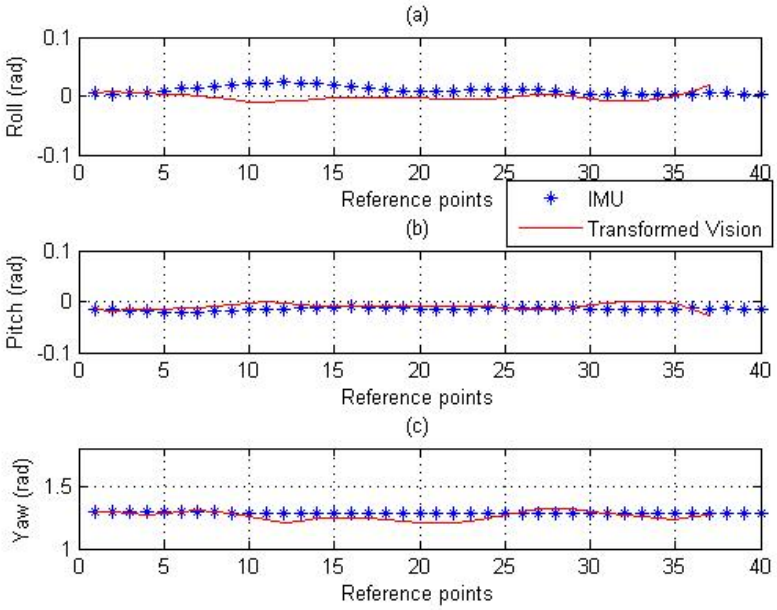


Figure 46: Comparison of raw inertial data with transformed vision data for (a) roll, (b) pitch, and (c) yaw for the straight section (longer run).

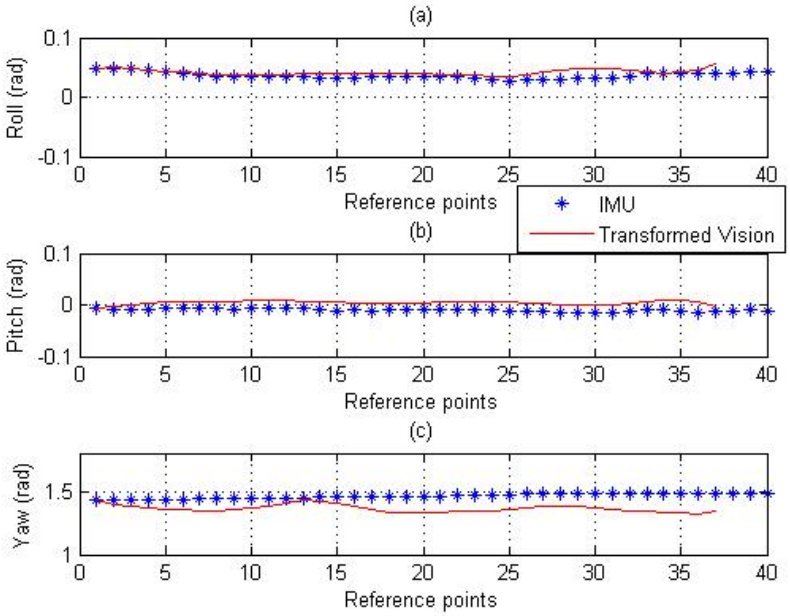


Figure 47: Comparison of raw inertial data with transformed vision data for (a) roll, (b) pitch, and (c) yaw for the horizontal curve (longer run).

7.3.3 Verification with Ground Truth

Five intermediate points of the shorter segment spaced at 11.5 ft were demarcated and taped in such a way that the vehicle would undergo translations in all three directions, and rotations about all three axes (roll, pitch, and yaw) between each consecutive locations. The manual survey was performed using a total station which could capture all the locations from one temporary benchmark. At each taped location, four separate points in the cross section of the pavement, edge, center, and two points in between, were surveyed. Moreover, at each of these points, total station measurements were repeated in order to eliminate any possible errors in the line of collimation. By considering the first surveyed point as the reference, horizontal and vertical angles between each pair of measurement points were estimated. From these measurements roll, pitch, and yaw of the vehicle at all consequent measurement points could be estimated with respect to the first point. The above survey measurements were then compared with the transformed vision Eqn (6-15) and transformed inertial measurements Eqn (6-14).

Since the optimized transformation between inertial and vision systems is available ($\mathbf{R}_{vi}$) the required transformations from inertial-survey Eqn (6-14) and vision-survey Eqn (6-15) can be obtained as shown in Chapter 6. Comparisons of these two transformed measurements with survey measurement are illustrated in Fig. 48. From Fig. 48 it is clear that the survey measurements and the two transformed measurements have a reasonably good agreement. The discrepancies can again be attributed primarily to the coarseness of data measurement interval, which cannot be lowered below 5 ft due to practical

difficulties. This affects the accuracy of determining the vision *pose* in particular (Chapter 4).

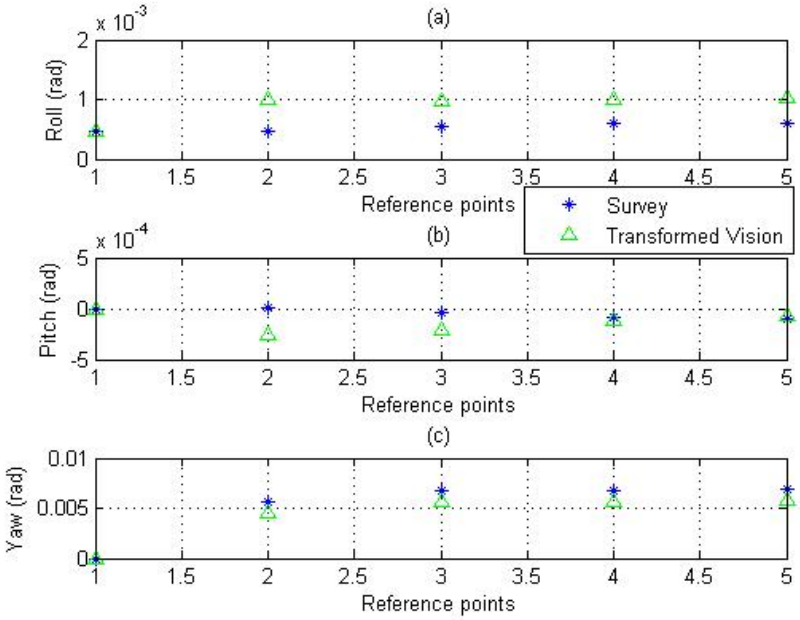


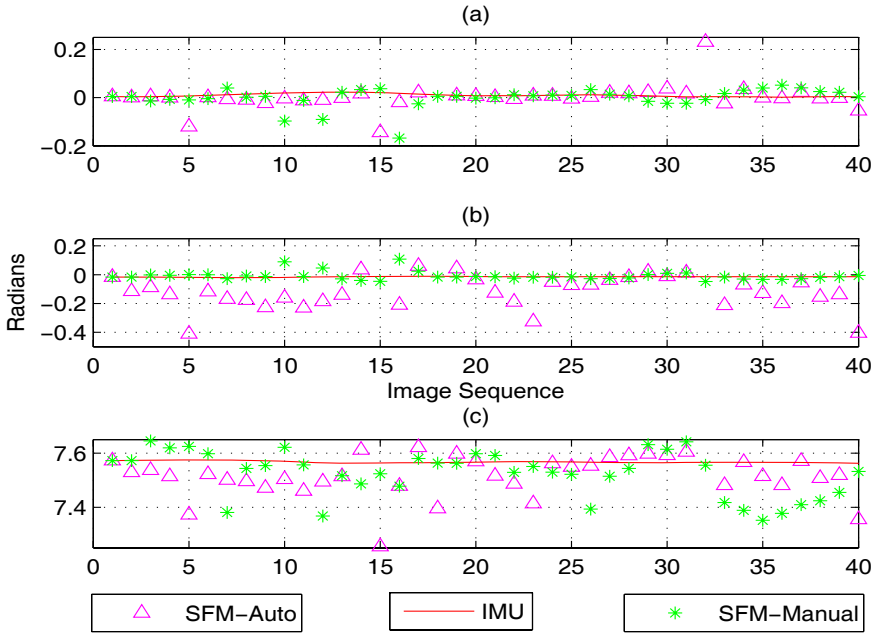
Figure 48: Comparison of transformed inertial and transformed vision data with survey data for (a) roll, (b) pitch, and (c) yaw on the short section.

7.4 Comparison of the Two Vision *pose* Measurement Systems

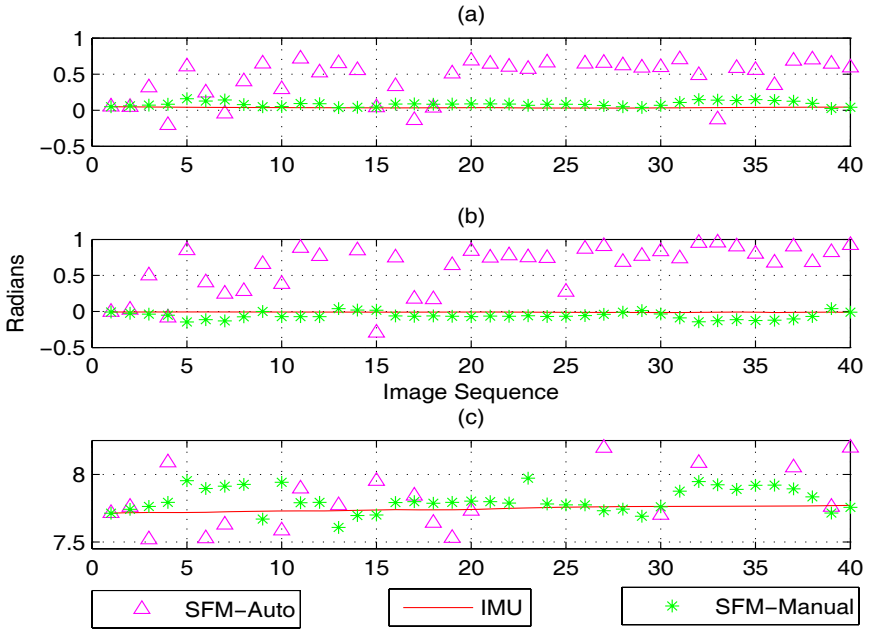
The SFM algorithm provides two outputs. They are (i) the rotation of the camera coordinate frame at the capturing instant of a given image with respect to that of the capturing instant of the next image, and (ii) normalized translation of the vehicle between the above two instances. It is noted that due to the scale factor associated with the depth direction, only the normalized (unit) vector can be provided for translations. The rotation angles can be expressed in Euler form enabling them to be compared with IMU rotations. The normalized translations cannot be directly compared with the translations derived from the IMU since the latter contains both directions and magnitudes. Therefore, in

order to compare the IMU translations with those of SFM, the former ones have to be normalized.

Figs. 49 (A-B) shows the comparison of orientations obtained from three different methods, IMU, manual SFM and auto SFM, plotted on the IMU frame for the two test sections. It can be noted from Figs. 49(A-B) that orientations derived from both SFM algorithms generally agree with the orientations obtained from IMU system. Moreover, it is also clear that the orientations predicted from manually picked correspondences, i.e. manual SFM, are closer to the IMU orientations than those predicted from the automated SFM algorithm where the correspondences are automatically identified.



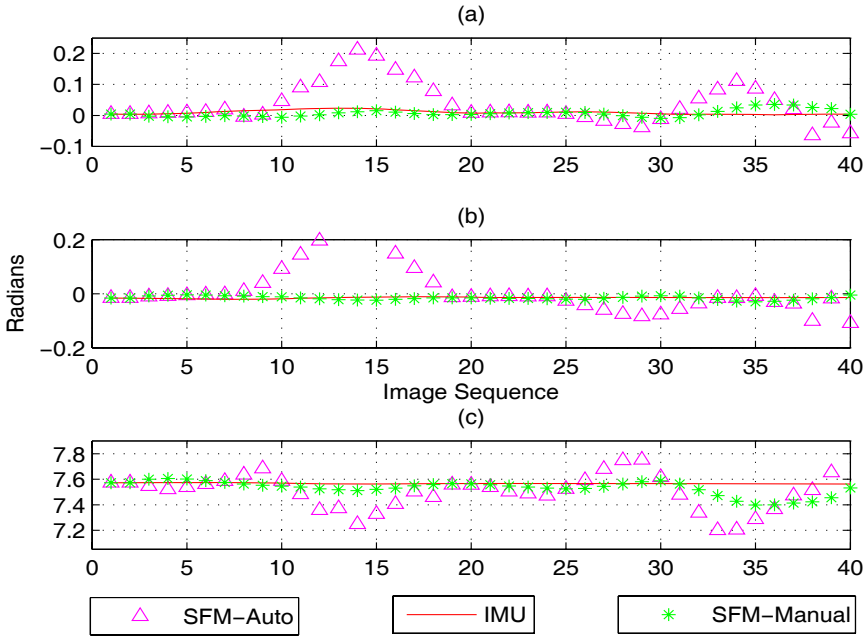
(A)



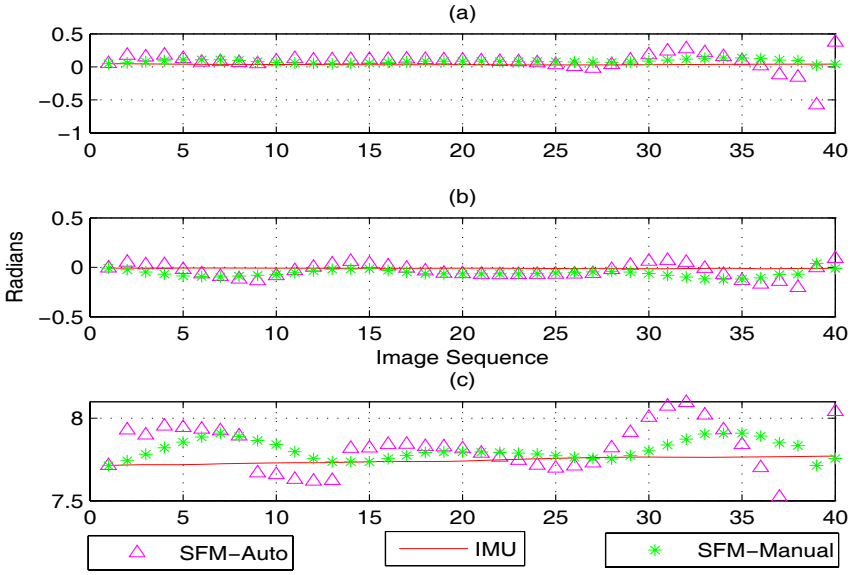
(B)

Figure 49: Comparison of orientations on the (A) straight section, (B) horizontal curve for (a) roll, (b) pitch and (c) yaw.

In order to reduce the variability associated with the SFM predictions and remove possible outliers inherent in the orientation estimation, first a median filter and then a moving average filter can be used. The raw orientation data can be processed with a median filter first to reduce the effects of the outliers and then the moving average filtration is performed on the refined data to reduce the fluctuations. The processed orientations data are plotted in Fig. 50 (A-B).



(A)



(B)

Figure 50: Comparison of orientation after moving median and moving average filtering for (A) straight section, (B) horizontal curve (a) roll, (b) pitch and (c) yaw.

Fig. 51 illustrates the comparison of the normalized translations obtained from the SFM algorithm and the IMU system. From Fig. 51 it can be seen that the raw translations obtained from the automated SFM have somewhat of a deviation from the IMU predictions. However, Fig. 51 also illustrates how the SFM translations processed by the Kalman filter algorithm match well with the IMU predictions.

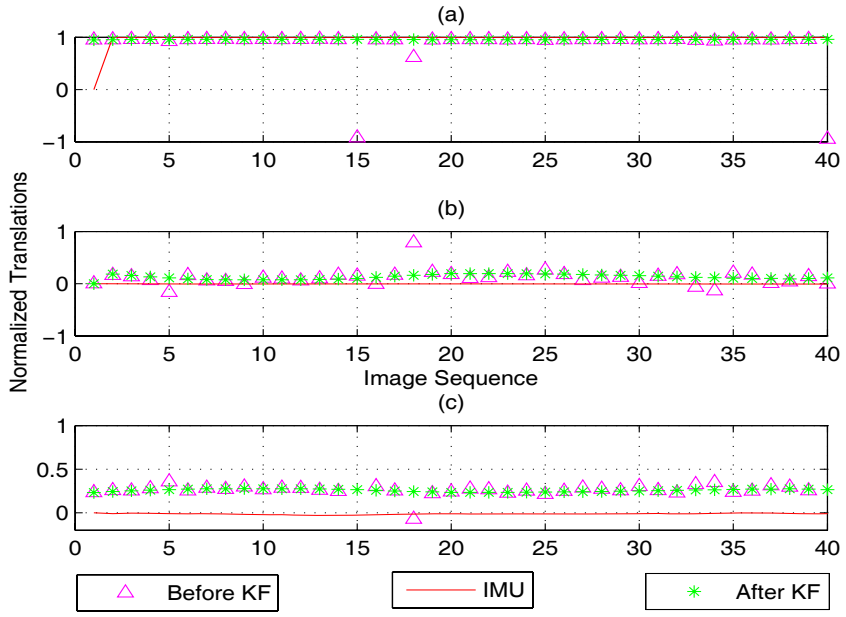


Figure 51: Comparison of normalized translations for the straight run before and after Kalman filtering (a) x-direction, (b) y-direction and (c) z-direction.

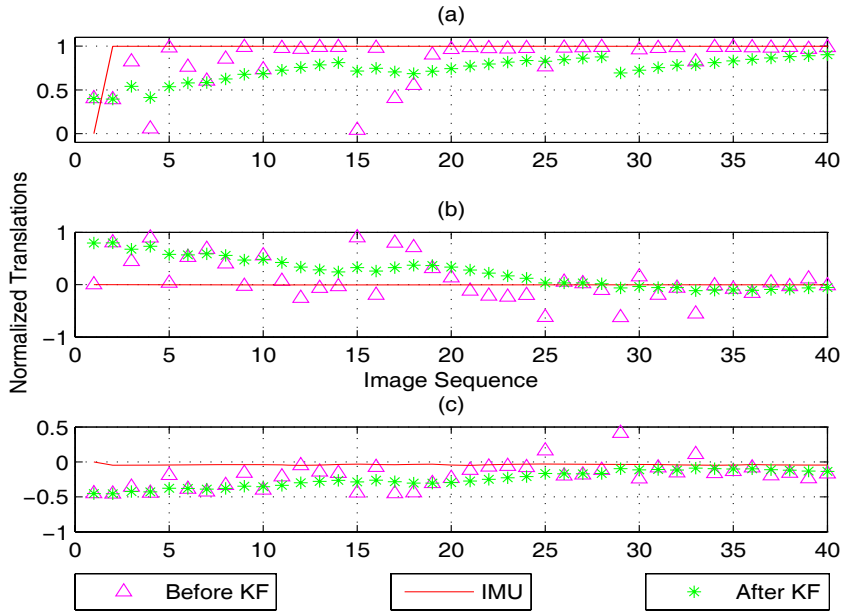


Figure 52: Comparison of normalized translations for the horizontal curve before and after Kalman filtering (a) x-direction, (b) y-direction and (c) z-direction.

It is clear from Figs. 49-52 that the rotations predicted from manual correspondences yield more accurate results than its automatic counterpart. The primary reason for the higher accuracy could be attributed to the fact that the manual correspondence algorithm uses straight lines in estimating the rotations while in the automatic algorithm discrete points are used. It is realized that matching corresponding points in two consecutive images is much more tedious than matching corresponding straight lines in images. In addition, the manual correspondences identified by a human would be less prone to errors compared to the automated correspondences assigned using a computer program.

7.5 Results of the IMU/Vision Integration

The translations and rotations of the test vehicle were estimated from vision sensors using the point correspondences tracked by the KLT tracker on the both sections. In order to

estimate the *pose* from the vision system, the correspondences given in Fig. 20, filtered out using the two methods mentioned in Chapter 5, were used. Figures 53 (a)–(c) compare the orientations obtained from both the pre-filtered vision system and the IMU/GPS system.

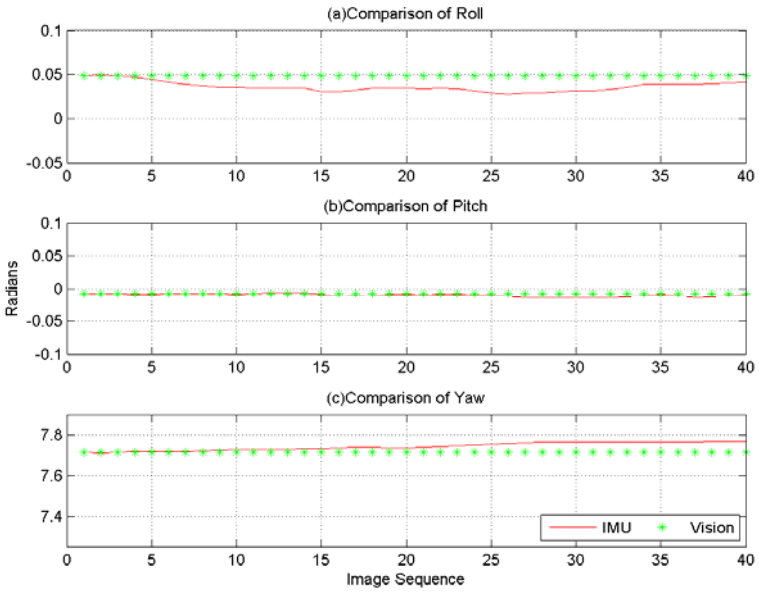


Figure 53: Comparison of (a) roll, (b) pitch and (c) yaw of IMU and filtered Vision

Similarly, the normalized translations are also compared in Figures 54 (a)–(c).

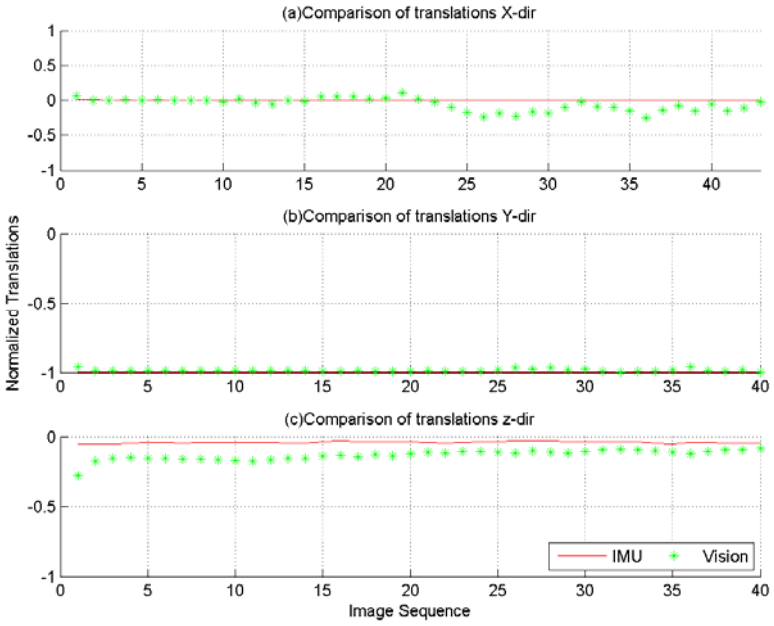
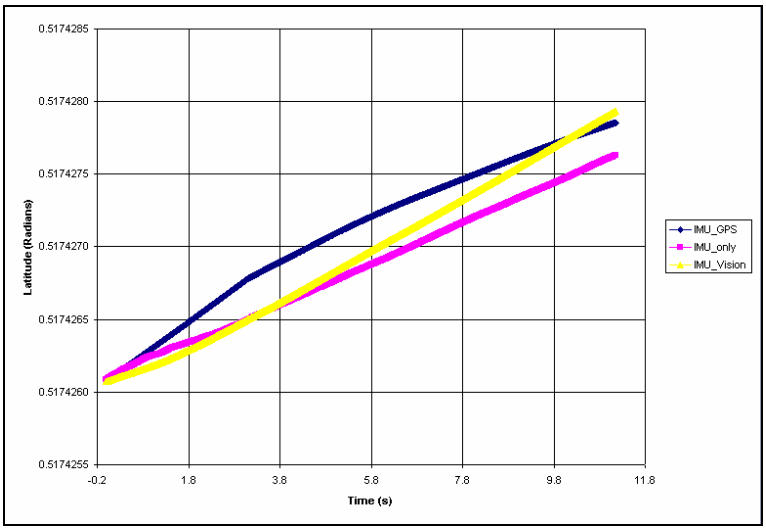
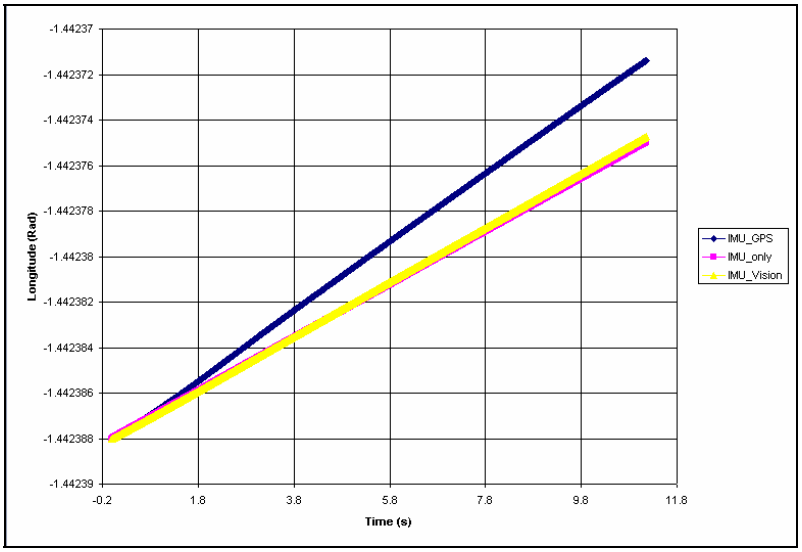


Figure 54: Comparison of translations (a) x-direction, (b) y-direction and (c) z-direction

It is clear from Figs. 53 and 54 that the orientations and normalized translations obtained by both IMU/GPS and filtered vision system match reasonably well. Hence, the authors determined that both sets of data are appropriate for a meaningful fusion and upgrade. These data were then used in the fusion process to obtain positions shown in Figs.55-a, b;



(a) Latitude



(b) Longitude

Figure 55: Comparison of (a) Latitude and (b) Longitude.

Furthermore, it is also clear from Fig. 55 that the IMU/Vision system estimates are much closer to those of the IMU/GPS system than the corresponding estimates of the IMU-only system. This is clearly shown in the latitude comparison (Fig 55a) where the IMU-only estimate consistently deviates from the IMU/GPS readings whereas the IMU/Vision

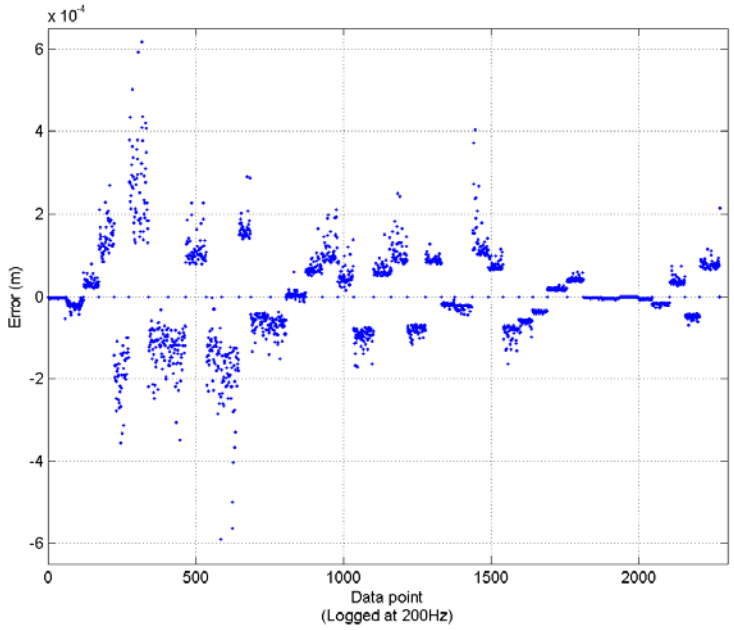
estimates approaches the latter after the initial deviation. The comparison results are summarized in Table 9.

Table 9: Maximum and minimum errors between IMU/GPS, IMU-only and IMU/Vision systems.

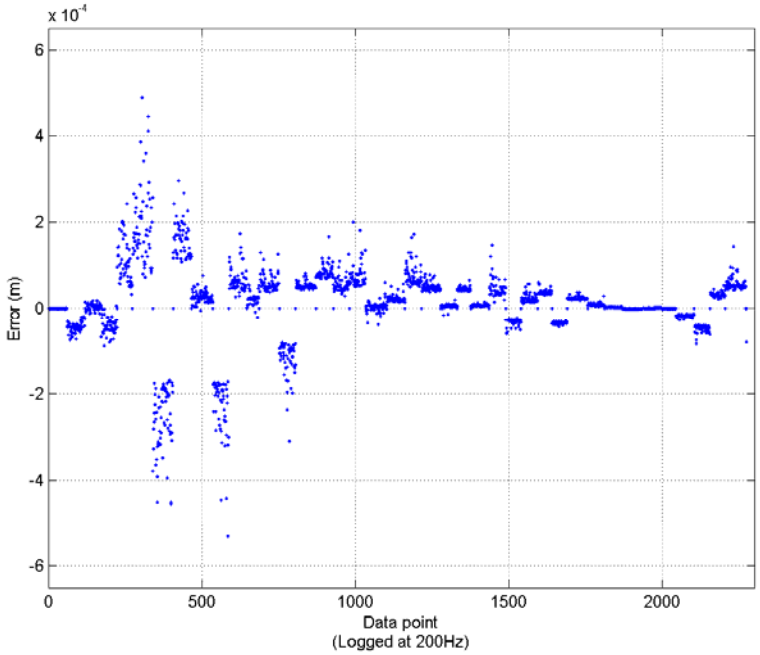
	IMU/GPS	IMU/Vis	IMU/Vision-IMU/GPS Error	IMU only	IMU only - IMU/GPS Error	Error (%) between errors estimated in Cols, 4, 6
Latitude	0.517426	0.5174266	2.824E-07	0.5174269	3.40E-07	20.38
Longitude	-1.442371	-1.4423747	3.354E-06	-1.442375	3.59E-06	7.02

Table 9 summarizes the two errors associated with both the IMU/Vision system and the IMU-only system with respect to the IMU/GPS system. It is clear that the IMU-only data consistently deviates from the IMU/GPS system due to IMU's inherent error growth. The last column of Table 9 indicates that at the end of the run, the respective latitude and longitude estimates of the IMU/Vision system are 20% and 7% closer to the IMU/GPS system than the corresponding estimates of IMU-only system.

The error estimated from the Kalman filter can be given as,



(a) Error in Latitude



(b) Error in Longitude

Figure 56: Error associated with (a) Latitude and (b) Longitude.

Figures 55-56 and Table 9 show that the position, i.e. latitude and longitude, estimated by the IMU/Vision integration agree quite well with that given by the IMU/DGPS integration. These results also clearly show that the IMU/Vision system can certainly supplement the IMU measurements without a significant loss of accuracy during a GPS outage. Furthermore, the authors have investigated the error estimation of the IMU/Vision fusion algorithm in Fig. 56. Figure 56 shows that the Kalman filter used for fusion achieves convergence and also that the error involved in the position estimation reduces with time. These results are encouraging since it further signifies the potential use of the vision system as an alternative to GPS in updating IMU errors.

7.6 Validation of IMU/Vision Orientation Results with Ground Truth

The survey for this validation can also be performed as previously discussed Section 7.1. These survey measurements were then compared with the IMU/Vision system orientations transformed into the appropriate frame to be compared with survey readings. This transformation can be found in (Randeniya 2006). Figure 57 illustrates the comparison between the IMU/DGPS, IMU/Vision and survey orientations for the short test section.

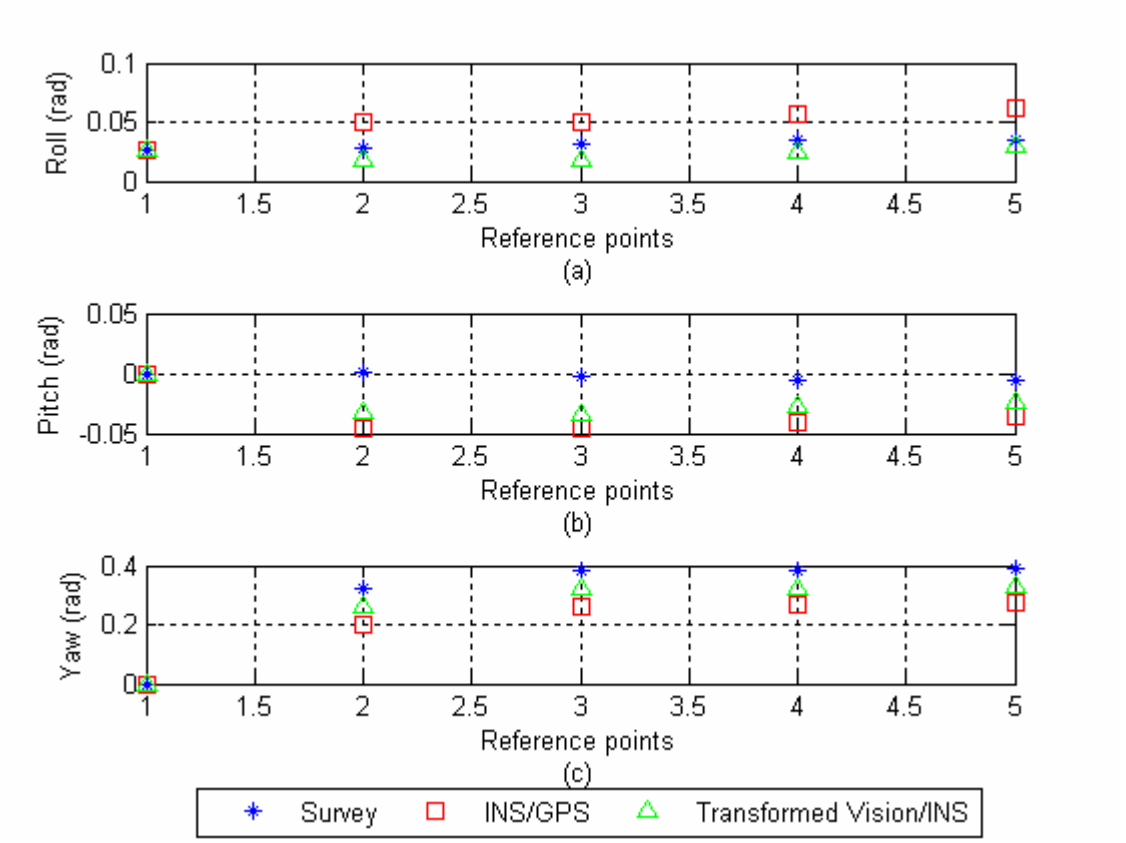


Figure 57: Comparison of IMU/Vision orientations with survey data.

It is clear from Fig. 57 that the IMU/Vision orientations are relatively closer to the survey orientations than those of IMU/DGPS. Since these results are obtained from a single test on an actual roadway one must be cautious and employ further experimentation to generalize the above conclusion. Table 10 shows the maximum error percentages estimated between the IMU/Vision, IMU/GPS systems and the survey. It is clear that from the Figure 57 and Table 10 that the orientations match reasonably well. The maximum percent errors are quite satisfactory considering the relatively large distance interval at which the images are captured.

Table 10: Maximum error percentages between the IMU/Vision and IMU/GPS system estimates and the actual survey.

Rotation	Survey estimate (rad)	IMU/Vision		IMU/GPS	
		Estimate (rad)	Error (%)	Estimate (rad)	Error (%)
Roll	0.0347	0.0296	14.71	0.051	46.97
Pitch	-0.018	-0.023	28.01	-0.030	66.67
Yaw	0.392	0.332	15.31	0.273	30.36

Finally, Table 10 compares the orientation measurements performed by IMU/Vision and IMU/GPS systems with respect to those of the actual survey. It is seen that for this particular test run, IMU/Vision system estimates orientations that are closer to survey measurements than those measured from the IMU/GPS system.

CHAPTER 8 – CONCLUSIONS

The work presented in this work primarily addresses two important issues involved in the process of fusing vision and inertial sensors; (*1*) estimating the intrinsic properties of the vision sensor (camera) and determining the optimized transformation between the inertial sensor and the vision sensor in an outdoor setting *(2)* Fusion of the IMU sensor system and the vision system was performed in aiding the autonomous navigational tasks as an alternative system to IMU/GPS system. Two validations were performed; *(1)* to compare transformed IMU/Vision measurements with IMU/GPS data, *(2)* to match transformed IMU/Vision measurements with global reference data (manual survey). The validation results show that the transformed vision measurements match reasonably well in both cases.

It was shown in this work that a vision system attached to a vehicle can be used to estimate the rotations and translations of that vehicle using a sequence of images. The results also showed that the vision data can be used successfully in updating the IMU measurements to address the inherent error growth. The fusion of IMU/Vision measurements was performed for a sequence of images obtained on an actual roadway and compared successfully with the IMU/DGPS readings. The IMU/DGPS readings were used as the basis for comparison since the main task of this work was to explore an alternative reliable system that can be used successfully in situations where the GPS signal is unavailable. Also it was shown in Figs. 53-54 that the noisy vision

measurements could be successfully used in the fusion after being processed by a pre-filter. This resulted in a successful IMU/Vision fusion effort in absolute position coordinates as shown in Fig. 55. In addition, convergence of the errors involved in Kalman filter within short time (Fig. 56) depicts the promise and effectiveness of the fusion algorithm. Furthermore, orientations obtained from the fused system were also successfully validated with a manual survey performed at the section. The author is confident that the accuracy of the IMU/Vision integrated system can be further improved by a closely spaced image sequence.

The author also found that the use of accurate correspondences is essential in executing the vision algorithm successfully. It was also seen that significant improvements can be made by employing special techniques such as epipolar lines and correspondence motion fields to eliminate the errors due to false correspondences. Therefore, the two (vision and inertial) sensor system can be used successfully to estimate the orientation of the vehicle to a considerable accuracy. However, some discrepancies do occur between the actual measurements and the predicted data. These discrepancies can be attributed to the large image capturing interval of the MPSV and can yield improved predictions once significantly finer intervals are used to collect the sequence of images from the vision sensor. It is clearly seen that this multi sensor fusion effort would certainly enhance the ITS and other artificial intelligence technologies and more importantly will present a more reliable form of navigation in GPS deprived environment.

8.1 Future Research

Due to the successful integration of IMU/Vision system in decentralized architecture, this work can be extended in following direction.

1) Fusing the IMU/Vision system in tightly coupled architecture.
2) Implementing the fused system in a land vehicle and use in real time navigation.

REFERENCES

[1] M. Cramer. (2005, December). GPS/INS Integration. [Online]. http://www.ifp.uni-stuttgart.de/publications/phowo97/cramer.pdf.

[2] M. Wei, K. P. Schwarz, "Testing a decentralized filter for GPS/INS integration", Position Location and Navigation Symposium, *IEEE Plans,* March 1990.

[3] S. Feng, C. L. Law, "Assisted GPS and its Impact on Navigation Transportation Systems," in *Proc5th IEEE International Conference on ITS*, Singapore, September 3-6, 2002.

[4] J. Farrell, M. Barth, "Integration of GPS/INS and Magnetic Markers for Advanced Vehicle Control," *PATH-UC Berkeley*, Report No. UCB-ITS-PRR-2002-32, Oct., 2002.

[5] Y. Zhao, "Mobile Phone Location Determination and its Impact on Intelligent Transportation Systems", *IEEE Trans. On ITS*, Vol. 1, No. 1, March 2000.

[6] P. Kumar, S. Ranganath, H. Weimin, K. Sengupta, "Framework for Real-Time Behavior Interpretation from Traffic Video," *IEEE Trans. On ITS*, Vol. 6, No. 1, March 2005.

[7] K. Sun, G. Arr, R. Ramachndran, S. Ritchie, "Vehicle Reidentification Using Multidetecor Fusion", *IEEE Trans. On ITS*, Vol. 5, No. 3, September 2004.

[8] U. Franke, S. Heinrich, "Fast Obstacle Detection for Urban Traffic Situations", *IEEE Transactions on ITS*, Vol. 3, No 3, Sep 2002.

[9] F. Dellaert, C. Thorpe, "Robust car tracking using Kalman filtering and Bayesian templates," *Conference on ITS,* 1997.

[10] M. Sotelo, F. Rodriguez, L. Magdalena, "VIRTUOUS: Vision-Based Road Transportation for Unmanned Operation on Urban-Like Scenarios", *IEEE Trans. On ITS*, Vol. 5, No. 2, June 2004.

[11] W. Li, H. Leung, "Simultaneous Registration and Fusion of Multiple Dissimilar Sensor for Cooperative Driving", *IEEE Trans. On ITS*, Vol. 5, No. 2, June 2004.

[12] D. Diel, "Stochastic constraints for vision aided inertial navigation," M.S. thesis, Dept. Mechanical. Eng., Massachusetts institute of technology, Cambridge, MS, USA, 2005.

[13] J. Hespanha, O. Yakimenko, I. Kaminer, A. Pascoal, "Linear parametrically varying systems with brief instabilities: an application to vision/inertial navigation", *IEEE Trans. On Aerospace and Electronic Systems*, Vol. 40, No. 3, July 2004.

[14] B. M. Scherzinger, "Precise Robust Positioning with Inertial/GPS RTK," Proceedings of ION-GPS 2000, Salt lake city, Utah, September, 2000.

[15] C. J. Taylor, and D. J. Kriegman, "Structure and motion from line segments in multiple images," IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 17, No. 11, Nov. 1995, pp. 1021–1033.

[16] J. Canny, "A computational approach to edge detection," IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 8, Issue 6, Nov. 1986.

[17] Ma Y., Kosecka J., Sastry S., 'Motion Recovery from Image Sequence: Discrete viewpoint vs Differential viewpoint', *ECCV*, 1998.

[18] Birchfield S. (2006 November), KLT: An implementation of the Kanade-Lucas-Tomasi Feature Tracker, [online]. http://www.ces.clemson.edu/~stb/klt/.

[19] Faugeras O., 'Three Dimensional Computer Vision: A Geometric Viewpoint.', Second edition, MIT press, Cambridge, Ma, 1996.

[20] K. Shoemake, "Animating rotation with quaternion curves," in ACM SIGGRAPH computer graphics, vol. 19, No. 3, San Francisco, July 22-26, 1985.

[21] Randeniya D., Gunaratne M., Sarkar S. and Nazef A., "Calibration of Inertial and Vision Systems as a Prelude to Multi-Sensor Fusion", Under Review *Transportation Research Part C (Emerging Technologies),* Elsevier, June 2006.

[22] D. H. Titterton and J.L. Weston, "Strapdown inertial navigation technology," in IEE Radar, Sonar, Navigation and Avionics se*ries 5*, E. D. R. Shearman and P. Bradsell, Ed. London: Peter Peregrinus Ltd, 1997, pp. 24–56.

[23] Grewal M, Andrews A, "*Kalman Filtering Theory and Practice using MATLAB*", 2nd edition, John Wiley & Sons, NY, USA, 2001.

[24] B. K. P. Horn , "Closed-form solution of absolute orientation using unit quaternions," *Journal of the Optical Society of America*, vol. 4, pp. 629, Apr. 1987.

[25] J. Alves, J. Lobo, and J. Dias, "Camera-inertial sensor modeling and alignment for visual navigation," in *Proc. 11th International Conference on Advanced Robotics*, Coimbra, Portugal, June 30 – July 3, 2003.

[26] P. Lang, A. Pinz, "Calibration of hybrid vision/inertial tracking systems," in *Proc. 2nd Integration of Vision and Inertial Sensors*, Barcelona, Spain, April 18, 2005.

[27] E. Foxlin, and L. Naimark, "Miniaturization, calibration and accuracy evaluation of a hybrid self-tracker," in *Proc. 2nd IEEE and ACM International Symposium on Mixed and Augmented Reality*, Tokyo, Japan, October 7-10, 2003.

[28] B. M. Scherzinger, "Precise Robust Positioning with Inertial/GPS RTK," Proceedings of ION-GPS 2000, Salt lake city, Utah, September, 2000.

[29] J. Heikkila, "Geometric camera calibration using circular control points," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 22, No. 10, Oct. 2000.

[30] Z. Zhang. (2006, February). A flexible new technique for camera calibration. *Microsoft Research.* [Online]. http://research.microsoft.com/~zhang/Calib/.

[31] Beer F.P., Johnston E.R. Jr, *Vector Mechanics for Engineers-dynamics*, Mc Graw-Hill, New York, 1977.

[32] Scherzinger B.M., "Precise robust positioning with inertial/GPS RTK", Proceedings of ION-GPS 2000, Salt lake city, Utah, September, 2000.

[33] *POS/LV V3 Installation and Operation Guide*, Applanix Corporation, Ontario, Canada, 2000.

[34] *POSPac User Manual*, Applanix Corporation, Ontario, Canada, 2002.

[35] Van Dine C.P., Overturf J., "Route mapping and linear referencing with photolog geometric data", Proceedings GEOTEC event, Vancouver, BC, March 2003.

[36] Minnesota DOT, *Roadway Design Manual (Metric).*

[37] http://www.dot.state.mn.us/tecsup/rdm/metric/3m.pdf , October 1999, accessed on 07/13/2004.

[38] Glenon J.C., Loumiet J.R., "Measuring Roadway Curve Radius Using the Compass Method", January 2003.

[39] Greenspan, R. L. (1995). Inertial navigation technology from 1970–1995. Navigation: Journal of The Institute of navigation, 42(1), 165–185. Special Issue.

[40] Gebre-Egziabher, D., Powell, J. D., and Enge, P. K. (2001). Design and performance analysis of a low–cost aided dead reckoning navigation system. In International Conference on Integrated Navigation Systems, St. Petersburg, Russia.

[41] D. Bevly, J Ryu and J.C. Gerdes, "Integrating INS sensors with GPS measurements for continuous estimation of vehicle sideslip, roll, and tire cornering stiffness", *IEEE Trans. on ITS*, Vol. 7, No. 4, Dec. 2006.

[42] R. Gregor, M Lutzeler, M Pellkofer, K.H. Siedersberger and E.D. Dickmanns, "EMS-Vision: A perceptual system for autonomous vehicles", *IEEE Trans. on ITS*, Vol. 3, No. 1, March 2002.

[43] M. Bertozzi, A. Broggi, A. Fascioli and S. Nichele, "Stereo vision-based vehicle detection", *Proc. IEEE Intelligent Vehicle Symposium*, 2000.

[44] A. Huster and S. Rock, "Relative position sensing by fusing monocular vision and inertial rate sensors," *Proceeding of the 11th International Conference on Advanced Robotics*, Portugal, 2003.

[45] Z Hu, U Keiichi, H. Lu H and F. Lamosa , "Fusion of vision 3D gyro and GPS for camera dynamic registration," *Proceedings of the 17th International Conference on Pattern Recognition*, 2004.

[46] M Bertozzi and A Broggi, "GOLD: A parallel real-time stereo vision system for generic obstacle and lane detection", *IEEE Trans. on image processing*, Vol. 7, No. 1, Jan. 1998.

[47] M Bertozzi, A Broggi and A Fascioli, "Vision-based intelligent vehicles: State of the art and perspectives", *Robotics and Autonomous Systems*, 32, Elsevier, 2000.

[48] J.C. McCall and M.M. Trivedi, "Video-based lane estimation and tracking for driver assistance: Survey, system, and evaluation", *IEEE Transactions of Intelligent Transportation Systems*, Vol. 7, No. 1, March 2006.

[49] S.I. Roumeliotis, A.E. Johnson and J.F. Montgomery, "Augmenting inertial navigation with image-based motion estimation", *Proceedings of IEEE on International Conference on Robotics & Automation*, Washington DC, 2002.

[50] J. Chen and A. Pinz, 'Structure and Motion by fusion of inertial and vision-based tracking', *Proceedings of the 28th OAGM/AAPR conference*, Vol. 179 of Schriftenreihe, 2004, pp 55-62.

[51] S. You and U. Neumann, 'Fusion of vision and gyro tracking for robust augmented reality registration', *IEEE conference on Virtual Reality*, 2001.

[52] E. Shin, "Estimation of techniques for low cost inertial navigation", *PhD dissertation*, University of Calgary, CA, 2005.

[53] C Jekeli, "Inertial navigation systems with geodetic applications", Walter de Gruyter GmbH & Co., Berlin, Germany, 2000.

[54] N. Barbour and G Schmidt, "Inertial sensor technology trends", *Trans. IEEE Sensors Journal*, Vol 1, No.4, Dec. 2001.

[55] D. Randeniya, M. Gunaratne, S. Sarkar and A. Nazef, "Calibration of Inertial and Vision Systems as a Prelude to Multi-Sensor Fusion", *Transportation Research Part C (Emerging Technologies)*, Elsevier, to be published.

[56] D. Randeniya, M. Gunaratne, S. Sarkar and A. Nazef, "Structure from Motion Algorithms for Inertial and Vision based Vehicle Navigation", under review by the Journal of Intelligent Transportation Systems, December 2006.

[57] D. A. Forsyth and J. Ponce, "Computer Vision A Modern Approach", Prentice Hall, NJ, USA, 2003.

[58] D.J. Allerton and H. Jia, "A review of multi sensor fusion methodologies for aircraft navigation systems", The Journal of Navigation, The Royal institute of Navigation, 58, 2005, pp 405-417.

[59] D.J. Allerton and A.J. Clare, "Sensor fusion methods for synthetic vision systems", *Proc. of the 23rd Digital Avionics Systems conference*, (DASC 04), Oct. 2004.

[60] L. Armesto, S. Chroust, M. Vincze and J. Tornero, "Multi-rate fusion with vision and inertial sensors", *Proc. of International conference on Robotics and Automation*, April 2004.

[61] M. Grewal, L. Weill and A. Andrews, "Global positioning systems, inertial navigation and integration", John Wiley & Sons, NY, USA, 2001.

[62] A.H. Jazwinski, "Stochastic Processes and Filtering Theory", Academic press, NY, USA, 1970.

[63] A. Mraz "Evaluation of Digital Imaging Systems Used in Highway Aplications", Doctoral Dissertation, University of South Florida, July 2004.

[64] Gunaratne, M., Mraz, A. and Sokolic I., "Study of the Feasibility Of Video Logging with Pavement Condition Evaluation", Report No: BC 965, July 2003.

[65] Gunaratne M and Randeniya D, "Use of an Improved Algorithm to Predict Road Curvature from Inertial Measurements", Program Report submitted to the Florida Department of Transportation, September 2004.

[66] Min Shin, Goldgof D, Bowyer K, "Comparison of Edge Detection Algorithms using a Structure from Motion Task", *IEEE Trans. On Systems, Man and Cybernetics-Part B: Cybernetics*, Vol.31, No 4, August 2001.

[67] Sharp C, Shakernia O, Sastry S, "A Vision System for Landing an Unmanned Aerial Vehicle", *Proceedings of the2001 IEEE International Conference on Robotics and Automation*, Seoul, Korea, May 2001.

[68] R.E. Kalman, “A new approach to linear filtering and prediction problem”, *Trans. On ASME Journal of Basic Engineering*, 82 (Series D), 35-45, 1960.

APPENDICES

Appendix A Structure from Motion Algorithms

A.1 Motion Estimation from Line Correspondences (Taylor et al., 1995)

Recovering the three dimensional structure of a scene with the aid of a moving camera is useful in estimating the translational and rotational vectors of the vehicle. Taylor et al. [19] introduced a methodology to estimate the Structure from Motion (SFM) from a scene composed of straight line segments, using the image data obtained by a camera attached to the system in motion. Extraction of the translational and rotational components from the captured images is achieved by minimizing an objective function in the form of;

$$O = \sum_{j=1}^{m} \sum_{i=1}^{n} Error(f(p_i, q_i), u_{i,j}) \tag{A-1}$$

Where,

$f(p_i,q_j)$: Function representing the image formation process.

p : Position of the three dimensional line.

q : Position and orientation of the camera.

$u_{i,j}$: Position and orientation measurement of the projection of feature i in image j.

Error ( ) : Positive real valued function that measures the difference between $u_{i,j}$ and $f(p_i,q_j)$.

The *Error* function in (A-1) can be expressed as;

$$Error = \mathbf{m}^T (\mathbf{A}^T \mathbf{B} \mathbf{A}) \mathbf{m} \tag{A-2}$$

Where, $\mathbf{m} = (m_x, m_y, m_z)$ is the surface normal to the image plane and

Appendix A (Continued)

$$\mathbf{A} = \begin{pmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \end{pmatrix};$$

$$\mathbf{B} = \left(\frac{l}{3(m_x^2 + m_y^2)} \right) \begin{pmatrix} 1 & 0.5 \\ 0.5 & 1 \end{pmatrix}.$$

(x_1, y_1) and (x_2, y_2) are the two end points (in terms of pixel coordinates) of any line correspondence in the image.

The minimization process mentioned above is a two stages, hybrid, minimization process consisting of global and local minimization components. To start the process, a random initial estimate for the camera orientation is selected. Once the initial estimates for the camera orientation is available the second stage of the minimization, local minimization, process starts. The latter is achieved in four stages.

If two coordinate frames are considered, i.e. the global coordinate frame denoted by *w* and the vision (camera) coordinate frame denoted by *c*, the perpendicular distance to any vector (v) measured in the camera coordinate frame (d^c) can be expressed as;

$$d^c = \mathbf{R}_w^c (d^w - t_c^w + (t_c^w . v^w) v^w) \tag{A-3}$$

Where $\mathbf{R}_w^c$ expresses the rotation between the two frames, t_c^w is the translation between the two frames measured in the global coordinate frame and d^w is the perpendicular distance to any vector (v^w) from the vision (camera) frame origin measured in the global coordinate frame. The surface normal to the image plane can be obtained as;

$$\begin{aligned} m^c &= v^c \times d^c \\ m^c &= \mathbf{R}_w^c \left\{ v^w \times (d^w - t_c^w) \right\} \end{aligned} \tag{A-4}$$

From (A-4) it is possible to deduce two constraints in the form of;

Appendix A (Continued)

$$m^{c^T}.(\mathbf{R}_w^c v^w) = 0 \tag{A-5}$$

$$m^{c^T}.(\mathbf{R}_w^c (d^w - t_w^c)) = 0 \tag{A-6}$$

From the constraint (A-5) it is possible to deduce the generalized optimization function;

$$C_1 = \sum_{j=1}^{m}\sum_{i=1}^{n}(\mathbf{m}_{ij}^T \mathbf{R}_j v_i)^2 \tag{A-7}$$

Where, $\mathbf{m}_{ij}$ is the evaluated normal to the plane passing through the camera center and the observed edge, $\mathbf{R}_j$ is the selected random camera orientation and v_i are the projection line directions. Therefore, once the random rotations are obtained at the global optimization step, (A-7) can be used to obtain the initial projection line directions. This is the first stage of the local minimization process. In the second stage the initial values obtained from the first stage are further optimized using (A-7).

In the third stage, the initial estimates for the camera translation and projection line positions are obtained. For this task a different objective function is deduced from the constraint given by (A-8);

$$C_2 = \sum_{j=1}^{m}\sum_{i=1}^{n}(\mathbf{m}_{ij}^T \mathbf{R}_j (d_i - t_i))^2 \tag{A-8}$$

Where d_i are the projection line positions and t_i is the camera translation.

Once the values for $\mathbf{R}_j$, v_i, d_i and t_i are obtained, the optimization function given in (A-1) is used to derive the optimized solution for both the structure of the scene and the position of the camera.

Appendix B Multi Purpose Survey Vehicles IMU and Camera Specifications

B.1 IMU Specifications (Northrop Grumman – LN 200)

LN-200
Fiber Optic Inertial Measurement Unit

Description

The LN-200 Inertial Measurement Unit (IMU) has been in high rate production since 1994. It is the latest in applied technology, utilizing state-of-art inertial fiber optic gyros and micro-machined accelerometers.

The LN-200 IMU is produced in a fully complemented and highly robotic production facility.

Applications

The LN-200 IMU has a wide variety of applications. Customers have purchased these products for space stabilization, missile guidance, Radar/EO/FLIR stabilization, motion compensation, UUV/UAV guidance and control, camera/mapping, and as IMUs for higher order integrated systems.

The LN-200 is a versatile inertial unit that is suitable for a wide variety of applications:

- Space Stabilization
- Camera/Mapping
- AHRS
- Motion Compensation
- EO/FLIR Stabilization
- Navigation
- Flight Controls
- ACMI/TSPI*

Advantages

The LN-200 is hermetically sealed and contains no moving parts or gaseous cavities, ensuring long, reliable shelf and usage life.

Northrop Grumman is the world's leading producer of inertial navigation systems with more than 45,000 systems in use worldwide.

Functionality

The LN-200, having been applied to a wide variety of applications, is available in a number of functional and data rate configurations.

* Air Combat Maneuvering Instrumentation/ Time, Space, Position Instrumentation

Appendix B (Continued)

LN-200
Fiber Optic Inertial Measurement Unit

LN-200 Features

- 3 solid-state fiber optic gyros
- 3 solid-state silicon accelerometers
- Miniature package <35 cu in. (88.9 cu cm)
- Lightweight package <2 lb (907.2g)

Heritage

LN-200 is on the following platforms:

- Clementine
- Satellites
- AGM-142
- Air Combat Maneuvering Instrumentation Pods
- BQM-74E
- GMLRS ATD
- LANTIRN
- Predator
- Global Hawk
- MK-48
- AMRAAM
- Stingray
- CH-46
- MB-339
- Radar – MoComp
- RAH-66

LN-200 Core IMU

Physical Characteristics	
Weight	<1.65 lb (750g)
Size	3.5 D x 3.35 H in. (plus connector) (8.89 D x 8.51 H cm)
Power	12W steady-state (nominal)
Cooling	Conduction to mounting plate
Performance—Accelerometer	
Bias Repeatability	300 µg to 3.0 milli-g, 1σ
Scale Factor Accuracy	300 to 5,000 ppm, 1σ
Performance—Gyro	
Bias Repeatability	1°/hr to 10°/hr, 1σ
Scale Factor Accuracy	100 to 500 ppm, 1σ
Random Walk	0.07 to 0.15°/sq rt hr Power Spectral Density (PSD) level
Operating Range	
Angular Rate	Up to ±11,459°/sec
Angular Acceleration	±100,000°/sec/sec
Acceleration	>70g
Angular Attitude	Unlimited
MTBF	>20,000 hr
Input/Output	RS-485 serial data bus (SDLC)
Environmental	
Temperature	-54°C to +71°C continuous operation
Vibration	15g rms, 20-20,000 Hz @ PSD NTE 0.114g^2/Hz in any bandwidth
Shock	90g, 6 msec terminal sawtooth

Appendix B (Continued)

B.2 Forward View Camera Specifications (DVC Company)

DVC Company, Inc.

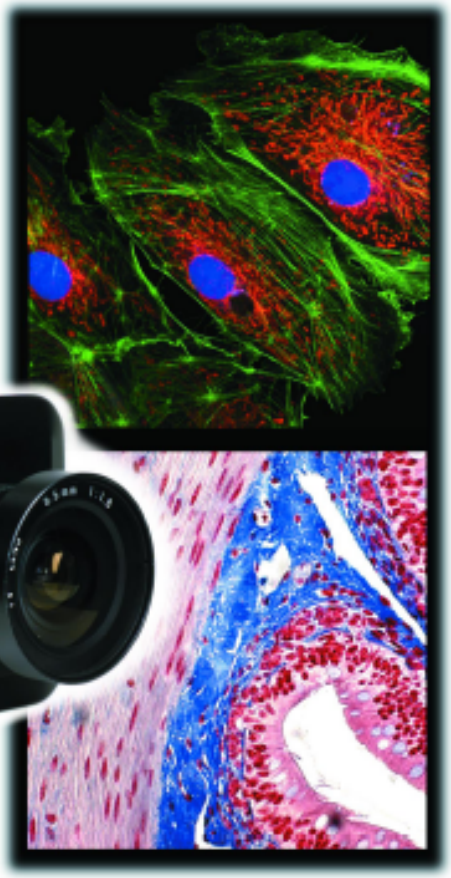
DVC-1310C-fw

Precision Megapixel Digital Color Camera

Features

- ***IEEE-1394 "FireWire" Interface for Mac and Windows***
- ***Megapixel Resolution:*** *1300 (H) x 1030 (V) x 10-bits*
- ***Fast Readout:*** *18.6 MSPS*
- ***High Sensitivity:*** *0.005 fc for full video at 11f/sec.*
- ***Long Term Exposure:*** *Up to 15 min.*
- ***High Signal / NoiseRatio:*** *60 dB*
- ***On-sensor Color Filter:*** *Bayer Pattern*
- ***RGB Digital Video Output:*** *10 bits/color/pixel*
- ***Progressive Scan:*** *11f/sec.*
- ***Image Capture and Control Software***
- ***Ultra-stable in-camera Digitization***
- ***Precision Opto-mechanical Mount***
- ***Optional TE Cooler:*** *-20°C without fan*
- ***No Mechanical Shutter required***
- ***CE/UL/CUL/FCC Certified***

Additional Information Available at:
www.dvcco.com



Description

The DVC-1310C-fw is a high resolution, RGB digital color camera utilizing a unique megapixel 2/3" progressive scan interline CCD sensor. The CCD sensor has a particularly high QE in the Blue-Green spectrum resulting in higher sensitivity for most applications.

RGB color is derived by use of an on-sensor Bayer filter pattern that minimizes light and resolution losses. The highly stable optical mount utilizes an adjustable C-mount coupling to provide critical system focusing adjustments.

The camera is supplied with both ***C-View*** and ***MAC-View*** for real-time viewing and image capture. ***C-View*** is a Windows based program which includes an integrated camera controller and viewer with both TWAIN and Image-Pro® drivers to facilitate ease-of-use with popular imaging programs. ***MAC-View*** is a Macintosh compatible plug-in for Photoshop®, QED Imaging and NIH Image-J. ***MAC-View*** provides real-time viewing and image capture within these programs.

The camera output and control are connected through a single 6-pin "FireWire" cable, which provides simple and reliable operation.

DVC-1310C-fw Product Brief Rev.1.1 July 2001

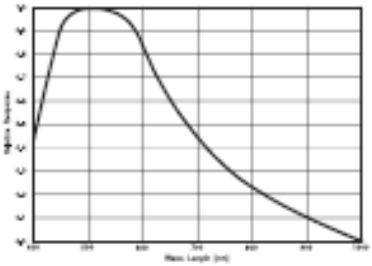
Appendix B (Continued)

Specifications

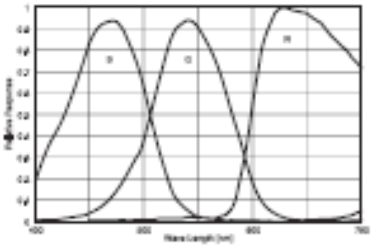
DVC-1310C-fw

Sensor:	**2/3" progressive scan interline CCD with Bayer filter**
Active picture elements	1300 (H) x 1030 (V)
Pick up area	8.71 mm x 6.90 mm (2/3" format)
Picture element size	6.7 µm x 6.7 µm (square format)
Aspect ratio	5:4
Digital Video output:	**10 Bit IEEE-1394A**
Readout rate	13.6 MHz, 11 frames/sec.
Frame Rate	11 frames/sec. @1300 x 1030
Signal to noise	> 60 dB, at min. gain
Sensitivity (light required for full scale output)	0.08fc@0dB gain, 11 f/sec 0.005fc@max gain,11 f/sec 0.001fc@0dB gain, 10 sec exposure
Gamma	1.0 (linear)
Camera Control:	**RS-232C, *C-View*™ interface software standard**
Gain control	30 dB
Offset control (black)	-16% to +34%
High speed shutter	87 µs to 91 ms
Integration control	91 ms to 15 min.
Electrical:	
Input voltage	110/220 VAC 50/60 Hz
Power	10 Watts
Clock	Internal crystal @ 74 MHz
Mechanical:	
Size	3.25" H x 3.25" W x 1.75" L
Weight (without lens)	12 oz. (424 grams)
Lens mount	Standard C-mount
Camera mount	1/4"x 20 Standard triped mount
Camera connector	6 pin Fire Wire
Power connector	DB-9M

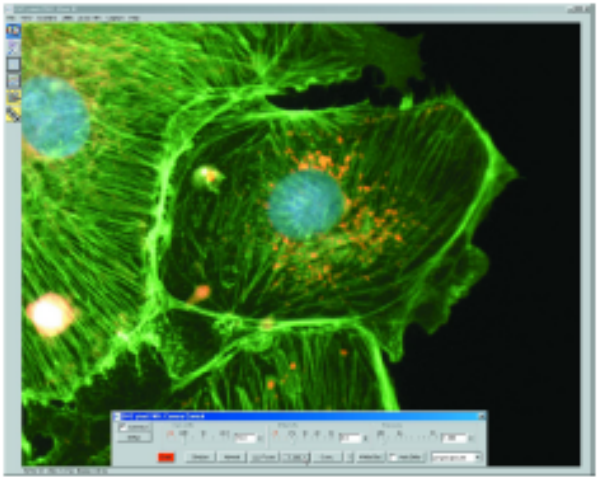
DVC-1312C-fw: Back View



CCD Spectral Response (without filters)



DVC-1310C-fw Spectral Response, Bayer mosaic filter



C-View™ Viewer and Camera Control Panel

Bayer mosaic pattern

ABOUT THE AUTHOR

Duminda Randeniya received his B.Sc. (Eng), with honors, in Mechanical Engineering from University of Peradeniya, Sri Lanka in 2001. He earned his Masters degree in Applied Mathematics from Texas Tech University in 2003 and enrolled in PhD program in 2004 at University of South Florida.

While pursuing the PhD, Mr. Randeniya actively participated in numerous research projects with Florida Department of Transportation. He also coauthored three journal papers and two conference papers.