Theses and Dissertations   Student Graduate Works

3-21-2013

Sensitivity Analysis of an Automated Calibration Routine for Airborne Cameras

Philip E. Lorenzini

Follow this and additional works at: https://scholar.afit.edu/etd

Part of the Other Electrical and Computer Engineering Commons

SENSITIVITY ANALYSIS OF AN AUTOMATED CALIBRATION ROUTINE FOR AIRBORNE CAMERAS

THESIS

PHILIP E. LORENZINI, CAPTAIN, USAF

AFIT-ENG-13-M-51

AIR FORCE INSTITUTE OF TECHNOLOGY

WRIGHT-PATTERSON AIR FORCE BASE, OHIO

The views expressed in this thesis are those of the author and do not reflect the official policy or position of the United States Air Force, Department of Defense, or the United States Government. This material is declared a work of the U.S. Government and is not subject to copyright protection in the United States.

AFIT-ENG-13-M-51

SENSITIVITY ANALYSIS OF AN AUTOMATED CALIBRATION ROUTINE FOR
AIRBORNE CAMERAS

THESIS

Presented to the Faculty

Department of Aeronautics and Astronautics

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

In Partial Fulfillment of the Requirements for the

Degree of Master of Science in Aeronautical Engineering

Philip E Lorenzini, MS

Captain, USAF

March 2013

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED

iii

SENSITIVITY ANALYSIS OF AN AUTOMATED CALIBRATION ROUTINE FOR
AIRBORNE CAMERAS

Philip E Lorenzini, MS

Captain, USAF

Approved:

_____   _____
Dr. John Raquet (Chairman)   Date

_____   _____
Dr. Jacob Campbell (Member)   Date

_____   _____
Karl Walli, Lt Col, USAF (Member)   Date

Abstract

Given a known aircraft location, a set of camera calibration parameters can be used to correlate features in an image with ground locations. Previously, these calibration parameters were obtained during preflight with a lengthy calibration process. A method to automate this calibration using images taken with an aircraft mounted camera and position and attitude data is developed. This thesis seeks to determine a partial set of circumstances that affect the accuracy of the calibration results through simulation and experimental flight test.

A software simulator is developed in which to test an array of aircraft maneuvers, camera orientations, and noise injection. The simulator uses a realistic aircraft model in order to accurately derive the inputs to the calibration routine. Features are generated and used to create a set of fictitious images to feed the calibration routine. Results from the simulation are used to prepare test points for an experiment flight test conducted to validate the calibration algorithm and the simulator.

Real world flight test methodology and results are discussed. Images of the ground along with precise aircraft navigation and time data were gathered and processed for several representative aircraft maneuvers using two camera orientations.

Only the straight and level maneuver is found to be detrimental to accurate calibration results. Feature measurement noise is found to be highly detrimental to parameter estimation while navigation noise has little affect. These results are validated with both simulated and experimental results.

Acknowledgments

I would like to acknowledge those who have helped me in the completion of this work. Thank you to Dr. John Raquet for giving a guy a chance and always having the time to sit down and give insight and mentorship. Thanks to the Have SURF test team, who toiled long hours in the name of a science. Most of all thank you to my wife, and sons who thanklessly stood tall during my long days and nights.

<div align="right">Philip E. Lorenzini</div>

Table of Contents

List of Figures

List of Tables

List of Abbreviations

SENSITIVITY ANALYSIS OF AN AUTOMATED CALIBRATION ROUTINE FOR AIRBORNE CAMERAS

I. **Introduction**

This thesis describes development, simulation, and flight testing of a routine to automatically determine intrinsic and extrinsic calibration parameters of an aircraft mounted camera. This work is a branch of research originated by Ohio University (OU) [**1**]. The refinement of the algorithm and development of the simulator was done at the Air Force Institute of Technology (AFIT). Experimental flight testing (project Have SURF [**2**]) was conducted at Edwards Air Force Base, CA as part of the USAF Test Pilot School curriculum.

Background

High precision calibration parameters are of great importance when cameras are used in aiding navigation. Additionally, the determination of accurate ground coordinates of a target or feature in an image is of high interest to the warfighter. Given a known aircraft location, a set of calibration parameters can be used to correlate features on an image to accurately determine ground locations. A series of these projections can be traced and input to a navigation computer to aid in position updates. Previously these parameters were obtained during preflight with a lengthy process accomplished before every sortie [**3**] [**4**]. Moreover, parameters derived from a ground based calibration do not account for the mechanical and thermal variations experienced by an aircraft structure in the air.

More recently, a method to perform this calibration using in-situ images of surveyed or generic ground features was developed by Ohio University [5]. Limited initial testing had shown viability of an automated calibration algorithm employing feature detection and correlation. No specific testing of aircraft maneuver, camera orientation and other factors which affect accurate calibration parameter estimation has been tested to date.

Problem Statement

As stated previously, methodology has been developed in which a set of camera calibration parameters can be extracted given a set of images correlated to an aircraft position and attitude. Given that the process uses a nonlinear least squares routine it was reasoned that changing the aircraft or camera orientation could affect the accuracy of a solution. Further, the inclusion of both noise in the feature detection process and in the navigation solution of the aircraft would lead to less than desirable estimation accuracy. This work seeks to answer the question of how aircraft maneuver, camera orientation, and noise will affect the accuracy of a camera calibration estimates.

Scope and Assumptions

The scope of this work is to provide research and results for cases both of simulated and real world data collected during flight test. The simulation is considered to be accurate only to the extent to provide known input data to the calibration algorithm. All experimental data was collected during actual flight conditions and considered to be accurate to the limits presented. Collected data was post processed to remove any

obvious imagery or data errors. A tertiary objective of the flight test was to gather and use data using lower fidelity sensors than those used in previous research.

Related Research

Using visual sensors to aid in the estimation of a navigation state is a rapidly evolving field. The accuracy of using images to estimate navigation parameters depends heavily on the accuracy of the calibration of the camera [3]. Traditionally camera calibration is a rather cumbersome process and must be done in the lab. Much research has been done in the field of camera self-calibration using a variety of methods and sensors [6] [7]. The research discussed here focuses on general camera aided navigation solutions, using available measurements from an INU to aid in determining the intrinsic and extrinsic camera parameters, and finally very relevant recent results from a research team at Ohio University.

Giebner.

The Air Force Institute of Technology has been at the forefront of the development of technologies coupling imaging sensors with navigation sensors. In 2002 Giebner [8] demonstrated the viability of an image aided inertial system via flight test at the USAF TPS. The tests proved that image aided inertial measurements was capable of providing near GPS levels of accuracy, however the processing relied on manual tracking and correlation of features across image sets. Further, a manual camera calibration procedure was employed by relating known angles in an experimental setup with measured angles from an image taken of a static scene of readily identifiable points. Though both of these

are effective techniques, man-made errors are inevitably induced when manual manipulation of data takes place.

Veth.

As a successor to the initial work of Giebner, Veth [3] sought to refine and automate the injection of image feature tracking to an onboard INS. Veth developed a self-contained optical inertial navigation system with automated feature tracking. The Scale Invariant Feature Transform (SIFT) was used to process images and determine feature sets. Furthermore, Veth successfully incorporated a camera calibration technique to account for both projection errors and radial optical distortion induced by the lens. The technique uses a rigorous nonlinear regression to extract the intrinsic properties of the cameras. In this way, the camera errors are able to be automatically fed into the filtering algorithm and accounted for. The downfall, however, is that the calibration parameters need to be determined prior to data collection to be properly implemented. Veth presents experimental results which validate both the validity of the optical aided inertial measurements and the camera calibration techniques.

Neilson.

Nielson's work is a follow on to the successful realizations of Veth [4]. Nielson focus was to perform a successful flight test to demonstrate the capability gained by integrating optical measurements with a low cost consumer grade IMU. Testing showed a marked improvement over the stand alone IMU. Also, further improvements in accuracy were obtained with the use of other sensor data available on board the aircraft.

Nielson was able to use the work of Zhengyou [9] to model the distortion of the camera as a sixth-order polynomial. This un-distort transform was then applied to the feature sets obtained with SIFT. Though promising, the calibration technique still requires extensive preflight calibration of the camera to determine the error model.

Luong and Faugeras.

Luong and Faugeras [6] provide a novel approach to solve for the intrinsic calibration parameters of a moving camera. By using point correspondences from successive images, the presented algorithm is able to determine the intrinsic calibration parameters of the camera, the relative displacement of the camera between images and the structure of the scene. Unlike previous methods, this approach only requires one input in order to produce the calibration parameters, relative displacements between the cameras and structure of the scene with no initial information about the camera. Furthermore, no knowledge of camera motion is needed. If this additional information was available more accurate results could be computed [6]. Though Loung and Faugeras present a method for online calibration, they do not address the camera extrinsic parameters; a requirement for an aircraft mounted sensor.

Mamatas, Mohamed, and Soloviev.

Mamatas, Mohamed, and Soloviev [7] propose a method of resolving intrinsic camera calibration parameters with the aid of measurements from an IMU. Properties of image epipolar geometry through translations and rotations of the camera are exploited to derive the essential and fundamental matrices. These matrices are then exploited to resolve the

intrinsic properties of the camera. Though this method is efficient and draws on the availability of IMU, measurements assume a pinhole camera model as well as known extrinsic parameters between the camera and IMU.

Cohenor and Van Graas.

Cohenor and Van Grass [**1**] and [**5**]implemented a method of refining a set of both intrinsic and extrinsic calibration parameters of an aircraft mounted camera. Two methods are presented: a manual and an automated technic. The manual method relies on sets of ground targets with known surveyed locations combined with an accurate source of aircraft position and attitude. A linear least squares method is used to minimize the error between the known target location and the projected location given an initial set of calibration parameters. User input is required to manually designate the true location of the target in an image. A second, automated method is presented which used a feature detection algorithm in place of the user input. No measurement information regarding the detected ground targets is needed. This method used a nonlinear least squares estimation routine to calculate the calibration parameters as well as the location of the detected features. Cohenor and Van Graas' research, with changes as noted, is the basis for the investigation presented in this thesis

Summary

This thesis begins with a brief overview of optical imaging as well as scene reconstruction and projection. Current methodology for the extraction of camera calibration parameters is discussed along with the bundle adjustment technique used with

the airborne camera. The projection and calibration algorithms are discussed in depth and shown to be viable. The development and use of the simulation software is presented next. Finally, several sets of results are illustrated. The simulation software is used for several test cases. The experimental flight test methodology is presented as well as issues and results from the test. Finally, the simulation and flight test data are combined in order to gain further insight into the problem. Several supported conclusions are made and notes for future work are presented.

Background

Imaging and Camera Calibration

When used to derive a precise input for targeting or navigation the data provided by an imaging sensor has to be highly accurate. One of the largest error sources from a camera is due to the distortion created by the camera as it images a scene. A calibration process is used to account for these error sources and provide an accurate solution that relates scene points with image points.

Notation

The following notational conventions are used:

- *Scalars* – Upper or lower case letters in italic font (e.g. λ, x)
- *Vectors* – Bold lower case letters. (e.g. $\boldsymbol{p}$) Subcomponents of the vector are denoted with a subscript. The ith component of $\boldsymbol{p}$ is shown as $\boldsymbol{p}_i$. Vectors joining two points are denoted with an arrow (e.g. $\overrightarrow{\boldsymbol{po}}$)
- *Matrices* – Bold upper case letters (e.g. $\boldsymbol{P}, \boldsymbol{M}$)
- *Reference Frames* – Italicized capital letters (e.g.I, W). A vector in a specific frame is denoted with a superscript $\boldsymbol{s}$ in the world frame would be given as $\boldsymbol{s}^W$.
- *Transformations* – Italicized upper case letters with a super and a subscript. A transformation from A to B is given as T_A^B. If a transformation is done via a direction cosine matrix it is denoted with C_A^B.

Optical Perspective Projection.

The relationship between real world scene geometry to that of an image passed through a lens is described with optical projection theory. Image projection is most easily understood with analysis of the pinhole projection model. The pinhole projection model reduces the size of the lens to a theoretical point which allows only a single ray of light to pass through the image plane, the pinhole, and some point in the scene. In reality

this model cannot exist and the pinhole has a finite size and each point on the image plane is a collection of a cone of rays. The pinhole model does, however, allow for straightforward analysis of image plane geometry. The image created through a pinhole model is inverted, while a right side up virtual image can be thought of as the inverse of the projected image and is located in front of the pinhole at the same distance as the image plane. This shows that the apparent size of an object on the image plane depends on the distance the object is from the pinhole. The following pinhole camera model was based on the work presented in [**10**]. The geometry of the image can be described in terms of a coordinate system which originates from the pinhole center o and is at a distance f from the image plane. As in Figure 1, s denotes a point in a scene that is passed through the pinhole lens model and is denoted on the image plane by p'. Point s, o and p' are collinear and satisfy $\overrightarrow{op'} = \lambda\overrightarrow{os}$. Let (x, y, z) denote the coordinates of s in the scene and (x', y', f') denote the location of p' in the image plane where $f' = z'$ then

$$\lambda = \frac{x'}{x} = \frac{y'}{y} = \frac{z'}{z} \tag{1}$$

and

$$
\begin{aligned}
x' &= f'\frac{x}{z} \\
y' &= f'\frac{y}{z}
\end{aligned}
\tag{2}
$$

This relation can be used to describe the intrinsic and extrinsic parameters of a camera.
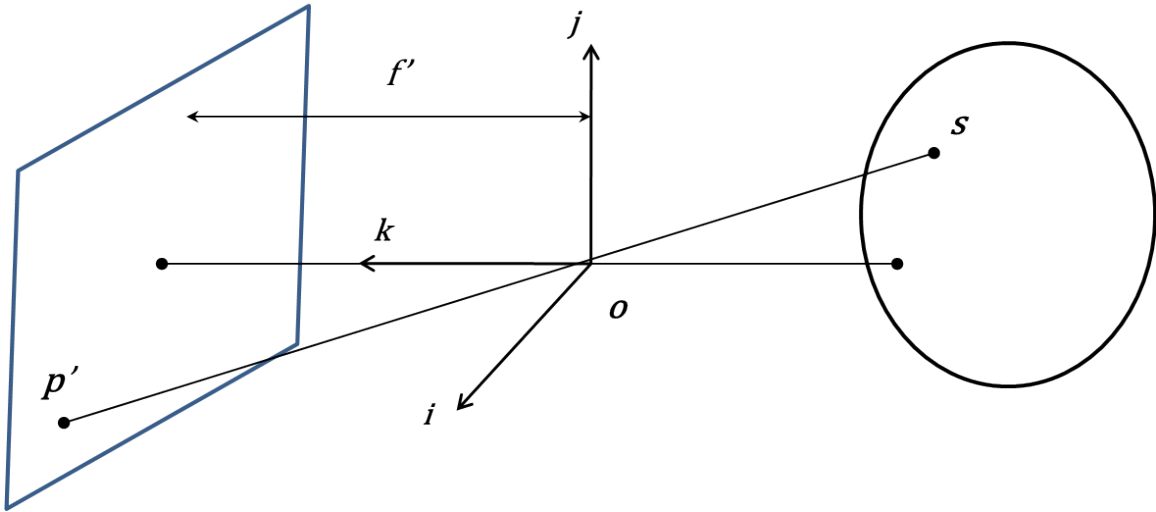
Figure 1. Perspective Projection Geometry

Camera Parameters.

The model presented above assumes that all distances are measured in the camera's reference frame and that the image coordinates have their origin at the principal point; the center of the camera's retina. In reality, the physical parameters of the camera should be taken into account to relate the world coordinates to that of the camera. The *intrinsic* parameters relate the camera's coordinate system to the idealized coordinate system in Figure 1, while the *extrinsic* parameters relate the camera's coordinate system to a world coordinate system.

When discussing the intrinsic parameters, it is easiest to start by assuming the properties of the pinhole model and neglect the focus and distortion effects associated with real lenses. To describe the intrinsic parameters, a normalized image plane is placed at a unit distance from the pinhole parallel to the cameras retina. This plane then has its own coordinate system with an origin located at point $\hat{c}$ as in Figure 2. With this, the relationship presented in equation two becomes

10

$$\hat{u} = \frac{x}{z} \\ \hat{v} = \frac{y}{z} \leftrightarrow \hat{p} = \frac{1}{z}[I \quad 0]\begin{bmatrix} s \\ 1 \end{bmatrix} \tag{3}$$

Where $\hat{p} = (\hat{u}, \hat{v}, 1)^T$ is the vector of coordinates of the projection $\hat{p}$ of the point p into

the normalized image plane. c_0
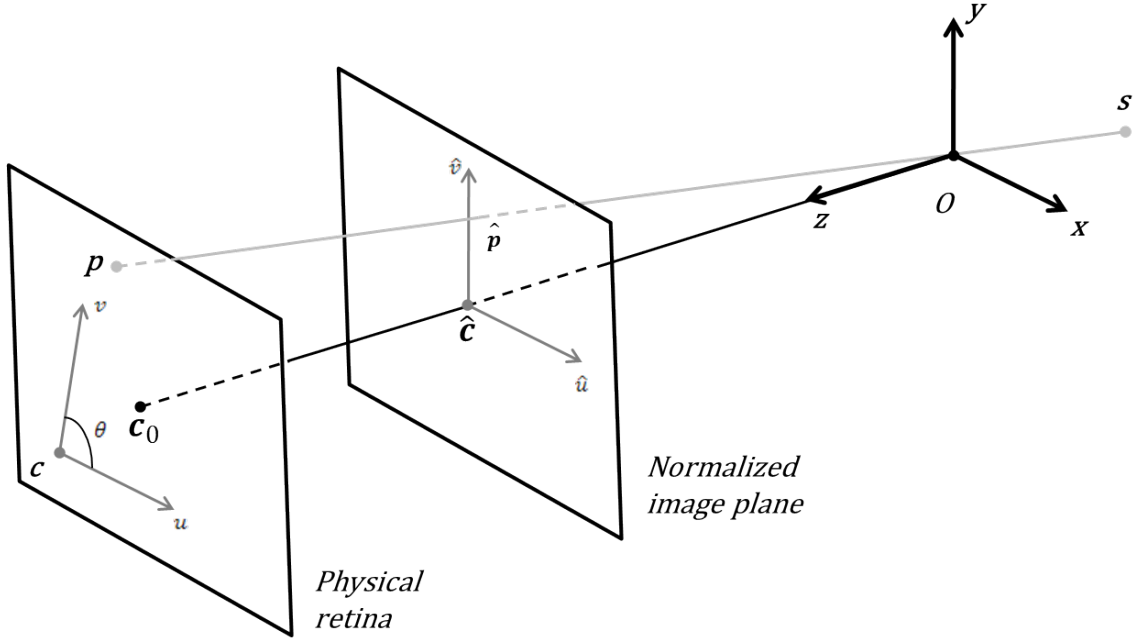
Figure 2. Physical and Normalized Image Coordinate Systems [10]

However, in general, the physical location of the retina is not located at a unit distance

from the pinhole. In addition, image coordinates are expressed in pixel units and are

sometimes rectangular in shape rather than square. This introduces additional scaling

parameters [10].

$$u = kf\frac{x}{z} \\ v = lf\frac{y}{z} \tag{4}$$

where f is the distance from the retina to the pinhole expressed in meters, and k and l are

scaling parameters expressed in pixels/meter. These parameters are not independent and

11

can be combined to form magnification parameters $\alpha = kf$ and $\beta = lf$ expressed in pixels. The camera coordinates system usually originates from a corner of the retina rather than the center, this adds two more parameters u_0 and v_0 that define the center of the retinal coordinate system [10].

$$u = \alpha \frac{x}{z} + u_0$$
$$v = \beta \frac{y}{z} + v_0 \tag{5}$$

The last correction is to the camera skew angle that is induced by some manufacturing error. In other words u and v are not exactly at ninety degrees but rather are related by an angle θ and equation five becomes

$$u = \alpha \frac{x}{z} - \alpha \cot(\theta \frac{y}{z}) + u_0$$
$$v = \frac{\beta}{\sin \theta} \frac{y}{z} + v_0 \tag{6}$$

By combining equations three and six a transformation matrix between the physical image plane and the normalized plane is formed.

$$p = K\widehat{p} \tag{7}$$

Where

$$p = \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \quad \text{and} \quad K = \begin{bmatrix} \alpha & -\alpha \cot \theta & u_0 \\ 0 & \frac{\beta}{\sin \theta} & v_0 \\ 0 & 0 & 1 \end{bmatrix} \tag{8}$$

and the final transformation becomes

$$p = \frac{1}{z} M\underline{p} \tag{9}$$

Where

$$\mathbf{M} = [\mathbf{K} \quad \mathbf{0}] \tag{10}$$

and $\underline{p} = (x, y, z, 1)^T$ is the coordinate vector of s in the camera coordinate system. This allows for the transformation of world coordinates to image point coordinates with the 3×4 matrix $\mathbf{M}$. Though the physical size of the pixels and skew are always fixed for a given camera, the values may not be readily available and must be accounted for through calibration.

The extrinsic parameters are used to relate the camera coordinates to the world coordinates through a rotation matrix $\mathcal{R}_w^c$ and a translation vector t with

$$\begin{bmatrix} p_c \\ 1 \end{bmatrix} = \begin{bmatrix} \mathcal{R}_w^c & t \\ \mathbf{0} & 1 \end{bmatrix} \begin{bmatrix} p_w \\ 1 \end{bmatrix} \tag{11}$$

Where p_c and p_w are the coordinates of the point p in the camera and world frame respectively. This is combined with the intrinsic parameter estimation equation (1) to form

$$p = \frac{1}{z} \mathbf{M}_e p \tag{12}$$

where

$$\mathbf{M}_e = \mathbf{K}[\mathcal{R} \quad t] = \begin{bmatrix} \alpha r_1^T - \alpha \cot \theta r_2^T + u_0 r_3^T & \alpha t_x - \alpha \cot \theta t_y + u_0 t_z \\ \dfrac{\beta}{\sin \theta} r_2^T + v_0 r_3^T & \dfrac{\beta}{\sin \theta} t_y + v_0 t_z \\ r_3^T & t_z \end{bmatrix} \tag{13}$$

And r_1^T, r_2^T, and r_3^T denote the three rows of $\mathcal{R}$ and t_x, t_y, and t_z are the coordinates of the vector t.

13

Multiple View Imaging

The information presented thus far allows for the projection and translation of a three-dimensional (3D) world coordinate to a two-dimensional (2D) image coordinate. If multiple images, each with a different view, of the same 3D point are available, information about the scene can be extracted. This section presents the case in which a scene is present in two views, and then expands further to the unconstrained case of n views.

Epipolar Geometry.

Epipolar geometry describes the geometric relationship between a world point s and it's set of image points p and p'. Each image point has its own corresponding optical center o and o'; this is shown more clearly in Figure 3, where the focal plane is again projected in front of the optic by convention.
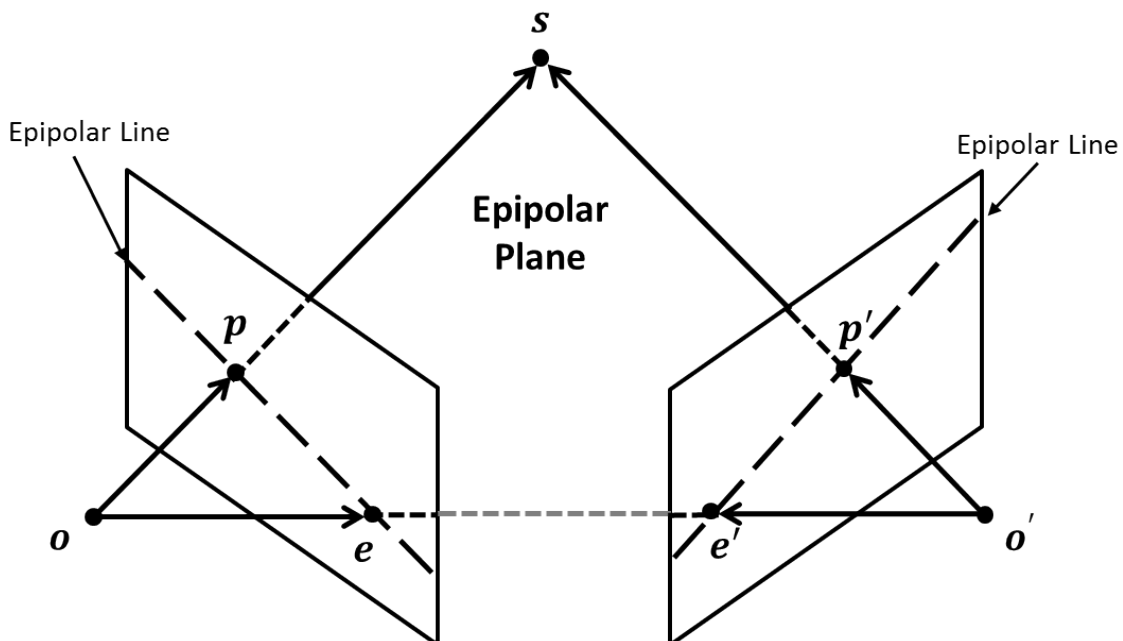
Figure 3. Epipolar Geometry

Five points constrain the epipolar plane, with points e and e' being the epipoles. Epipole e' is the projection of the optical center o of the first camera observed by the second camera and vice versa. If it is assumed that the intrinsic calibration parameters of each camera are known then a rotation and translation matrix can be defined to relate p and p'. This matrix is known as the essential matrix and has the property

$$p^T E p' = 0 \tag{14}$$

Where $E = [t_x]R$ is the essential matrix, $[t_x]$ is the skew-symmetric translation, and R is a rotation matrix. However, if the cameras are assumed to be uncalibrated a secondary relationship is formed. In this case $p = K\hat{p}$ and $p' = K'\hat{p}'$, where K and K' are the camera calibration matrices as in equation 11. The fundamental matrix F is the matrix which satisfies

$$F = K^{-T} F K'^{-1} \tag{15}$$

The fundamental matrix also has the property $p^T F p' = 0$ but is not in general an essential matrix [10]. The properties and derivations of the fundamental matrix are well documented and outside the scope of this paper; the reader is directed to [11] and [12] for further discussion. Many methods are available to extract a set of calibration parameters from up to four views [6], [11]. The following section will discuss the case of more than four view geometry with many point correspondences; three and four view geometry cases have been omitted because they do not apply to the approach described in this thesis.

Projective Reconstruction

As asserted above, any 3D s point can be projected through the camera matrix $\mathbf{M}$ to a 2D image point p. For simplicity the relation used here will be $p = \mathbf{M}s$. For the case of multiple views assume that n 3D points are visible in m images denoted by p_j^i; which is the projection of the ith point on to the jth image. Projective reconstruction involves finding the set of camera matrices $\mathbf{M}_j$ and points s_i that satisfy the projection across all images. Several algorithms exist to solve such a problem [**11**]; however the Bundle Adjustment methodology using a Levenberg-Marquardt nonlinear least-squares algorithm is the approach taken in this thesis.

Bundle Adjustment.

Bundle Adjustment, so named for the 'bundles' of light rays that form each image, seeks to mutually refine a set initial camera parameters and 3D point estimates to accurately predict the set of 2D image points p. This method is parameterized by reducing the squared distance between the set of predicted 2D points $\hat{p}$ and measured points p; minimizing the following cost function

$$\min_{\mathbf{M}_j s_i} \sum_{ij} d\left(\mathbf{M}_j s_i, p_{ij}\right)^2 \tag{16}$$

Where $d\left(\mathbf{M}_j s_i, p_{ij}\right)$ is the geometric distance between the predicted and measured image point. In general the camera parameters are assembled as a vector g_j and each 3D world point as vector s_i and compiled into a single vector $m = (g_1 \dots g_m, s_1 \dots s_n)$, while the measurements are combined into a single vector of the

form $p = (p_{11} \ldots p_{1m}, p_{21} \ldots p_{2m}, p_{n1} \ldots p_{nm})$.   Finally, given a initial parameter vector

$m_0 = (g_0, s_o)$ a projection function can be defined to estimate an initial set of 2D points,

$\hat{p} = f(m)$.   In this sense Bundle Adjustment amounts to minimizing $\epsilon = p - \hat{p}$ over m

and , which is most commonly done via the Levenberg-Marquardt Algorithm.

Levenberg-Marquardt.

The Levenberg-Marquardt (LM) algorithm is commonly employed to solve the

nonlinear least-squares minimization brought about by a bundle adjustment [**11**]. LM is a

combination of steepest descent and Gauss-Newton minimization methods.   If the

solution is far from a minimum the algorithm is slow but guaranteed to converge, much

like a steepest decent method.   As the solution approaches the local minimum the

algorithm becomes Gauss-Newton which converges quickly.   In-depth analysis of these

methods is beyond the scope of this thesis and the reader is invited to read [**13**] for more

detail.   As noted above we seek to minimize the quantity $\epsilon = p - \hat{p}$ for all s, LM does

this by taking an affine approximation for f in the neighborhood of s. In this sense f is

approximated by

$$f(m + \delta_p) \approx f(m) + J\delta_p$$

Where J is the Jacobian of f and $\|\delta_p\|$ is assumed to be small.   Each step in the LM

algorithm produces a vector m with the intent to minimize f and hence $\|p - f(m) -$

$J\delta_m\| = \|\epsilon - J\delta_m\|$. The key is to determine the step δ_m that produces $J^T(J\delta_m - \epsilon) = 0$.

This small step δ_p is then the solution to the linear least squares normal equation

$$J^T J \delta_m = J^T \epsilon$$

(17)

LM solves a slight variation of the normal equations known as the augmented normal equations

$$(J^TJ + \mu I)\delta_m = J^T \epsilon$$

(18)

The diagonal element of J^TJ are modified with a damping factor μ. In the algorithm δ_p is calculated along with the error vector $+\delta_m$. If this leads to a reduction in error the update is accepted, μ is reduced and the process is repeated. If the error increases, the damping term is increased and a new δ_m is repeatedly calculated until the error is decreased. The heart of the LM algorithm is repeatedly solved (18) using different damping factors until an acceptable error is obtained. An example of one form of the LM is presented in [**14**] and reiterated here for clarity.

RANSAC.

The RANdom Sample Consensus (RANSAC) allows for detection of outliers given two sets of data points [**11**]. RANSAC is particularly useful in the determination of the epipolar geometry between two images. Given two sets of points in successive images, RANSAC is used to determine an essential matrix which satisfies a collection of points while also identifying the outlying points for removal.

Input: $f(m)$, p, and m_0
Output: A vector m^+ minimizing $\|p - f(m)\|^2$
Algorithm:
$k = 0; \nu = 0; m = m_0;$
$A = J^T J; \quad \epsilon_m = p - f(m); g = J^T \epsilon_m;$
$stop = (\|g\| \leq \varepsilon_1); \mu = \tau * \max(A_{ii})_{i=1,\ldots,m}$
$while\ (not\ stop)and\ (k < k_{max})$
$\quad k = k + 1;$
$\quad repeat$
$\qquad Solve\ (A + \mu I)\delta_m = g;$
$\qquad if\ (\|\delta_p\| \leq \varepsilon_2(\|m\| + \varepsilon_2))$
$\qquad\qquad stop = true;$
$\qquad else$
$\qquad\qquad m_{new} = m + \delta_m;$
$\qquad\qquad \rho = (\|\epsilon_m\|^2 - \|p - f(m_{new})\|^2)/\delta_m^T(\mu\delta_m + g);$
$\qquad\qquad if\ rho > 0;$
$\qquad\qquad\quad stop = (\|\epsilon_m\| - \|p - f(m_{new})\|) < \varepsilon_4\|\epsilon_m\|);$
$\qquad\qquad\qquad m = m_{new};$
$\qquad\qquad\qquad A = J^T J; \quad \epsilon_m = P - f(m); g = J^T \epsilon_m;$
$\qquad\qquad\qquad stop = (stop)\ or\ (\|g\| \leq \varepsilon_1);$
$\qquad\qquad\qquad \mu = \mu * \max\left(\frac{1}{3}, 1 - (2\rho - 1)^3\right); \nu = 2;$
$\qquad\qquad else$
$\qquad\qquad\qquad \mu = \mu * \nu; \nu = 2 * \nu;$
$\qquad\qquad endif$
$\qquad endif$
$\quad until(\rho > 0)\ or\ (stop)$
$\quad stop = ((\|\epsilon_m\| \leq \varepsilon_3);$
$endwhile$
$m^+ = m$

Figure 4. LM Algorithm pseudocode [**14**]. Typical values for the stop

conditions are $\tau = 10^{-3}, \varepsilon_1 = \varepsilon_2 = \varepsilon_3 = 10^{-12}, \varepsilon_4 = 0, k_{max} = 100.$

Coordinate Systems

In order to expand the basic calibration algorithm to a real world setting, an understanding of coordinate systems is needed. Five different reference frames are used to describe the vectors and perspectives between the image, camera, and world.

Image Plane Frame (I-frame).

The image plane describes the location of a pixel in the image expressed in pixel units. The origin is defined as (1,1) and is located at the upper left pixel of the image. The x axis extends horizontally to the right from the origin and the y axis extended vertically downward. This is shown graphically in Figure 5.
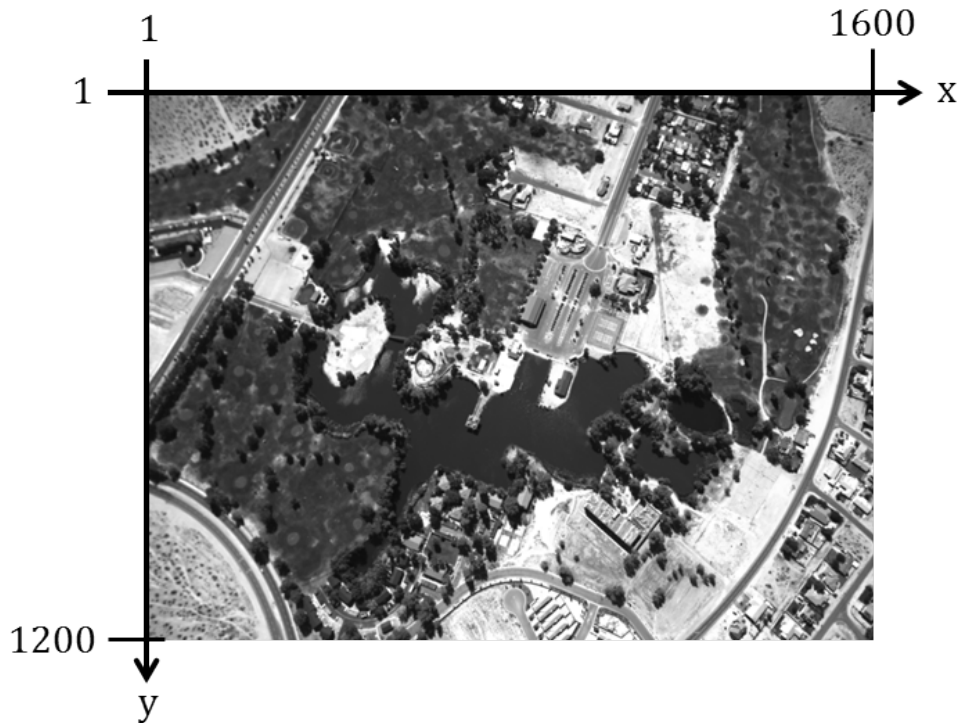
Figure 5. Image Plane Coordinate System.

Camera Frame (C-frame).

The camera frame has its origin at the optical center of the camera. The z-axis is the cross product of the x and y vectors as defined by the image plane. The z-axis extends from the camera's optical center out through the lens in the direction of the camera line of sight.

Navigation Frame(S-frame).

The aircraft navigation frame is centered at the navigation sensor located on the aircraft. The x-axis points out the nose of the aircraft, the y-axis point out the right wing, and the z-axis points vertically downward.

Camera Head Frame (H-frame).

The camera head frame describes the relationship of the camera rotation and translation to that of the aircraft navigation frame. The origin is located at the center of the navigation sensor. The z-axis points vertically downward along the same line of sight as the Camera Frame. The x-axis points out the nose of the aircraft and the y-axis point out the right wing.

World Frame (W-frame).

The world coordinate system is a local level coordinate system in the East, North, and Up directions and is derived from the Geodetic coordinates (Latitude, Longitude, Height) of the aircraft position. A common origin is used for all calculations and is fixed to a set of surveyed ground coordinates using the WGS-84 ellipsoid.

III.   **Automated Camera Calibration and Simulation**

This section describes a technique for camera calibration using position, attitude and image data collected during flight test. The process was initially presented in [**1**] and [**5**] and is included here for clarity and completeness with changes as noted. Two methods are available to calculate the camera calibration parameters: with and without knowledge of ground feature locations. The calibration method presented here does not use knowledge of ground feature world coordinates. The following sections describe the projection algorithm to relate image pixel locations to ground feature locations, the automated calibration algorithm, and software developed to test various aircraft trajectories and conduct a sensitivity analysis.

Pinhole Camera Projection Model

The projection from camera to world is a series of coordinate transforms, wherein each image pixel maps to a vector that defines a point in the scene. In this case the vector extends from the camera perspective center (CPC) through the image plane and is projected to a point on the ground. This projection is defined as a mathematical transformation from the image coordinate system I to the world coordinate system W, and is defined by T_I^W. The transform is dependent on the camera parameters $\boldsymbol{g}$, the camera pose $\boldsymbol{s}$ (attitude and position relative to the navigation frame), and a set of terrain elevation data $\boldsymbol{D}$. The transform is a cascade of transformation between frames and is presented as [**1**]

$$s = T_H^W(T_C^H(T_I^C(\boldsymbol{p}^I; \boldsymbol{g}); \boldsymbol{g}); \boldsymbol{g}, \boldsymbol{h}, \boldsymbol{D}) \qquad (19)$$

Where: $\boldsymbol{p}^C = T_I^C(\boldsymbol{p}^I; \boldsymbol{g})$ is the Image to Camera transform,
$\boldsymbol{p}^H = T_C^H(\boldsymbol{p}^C; \boldsymbol{g})$ is the Camera to Head transform,
$\boldsymbol{p}^W = T_H^W(\boldsymbol{p}^H; \boldsymbol{g}, \boldsymbol{h}, \boldsymbol{D})$ is the Head to World transform,

The camera model vector $\boldsymbol{g}$ is composed of the intrinsic and extrinsic parameters of the camera. In this model the intrinsic parameters include the CPC in pixels, and a second order model describing the lens distortion. The extrinsic parameters consist of the three Euler rotation angles, a translation vector that describes the camera head. The vector $\boldsymbol{h}$ contains the position and attitude of the aircraft and is obtained from a GPS/INS navigation computer. The final transformation to extend the world vector to intercept the ground requires information about the elevation of the terrain which can be obtained from a digital terrain elevation database (DTED), a series of discrete elevations or a single point. The final projection determines the projection of the image pixel location in the world frame by finding the intersection of $\boldsymbol{p}^W$ and the elevation $\vec{D}$.

Transformation from Image to Camera.

In reference to Figure 1 the vector $\overrightarrow{\boldsymbol{op}}$ extends from the CPC to pixel $\boldsymbol{p}$ in the image plane. Pixel $\boldsymbol{p}$ is located at $\boldsymbol{p}_x^I$ along the x-axis and $\boldsymbol{p}_y^I$ the y-axis. In this case the image origin is at (1,1) in the upper left corner, with the y-axis positive in the downward direction. The transform from Image to camera frame is given by

$$\boldsymbol{p}^C = T_I^C(\boldsymbol{p}^I; \boldsymbol{g}) = \begin{bmatrix} \boldsymbol{p}_x^I \\ \boldsymbol{p}_y^I \\ 0 \end{bmatrix} - \begin{bmatrix} \boldsymbol{g}_x \\ \boldsymbol{g}_y \\ \boldsymbol{g}_z \end{bmatrix} \qquad (20)$$

23

g_x and g_y are the x and y pixel locations of the center of the image plane, and g_z is the negative of the focal length in pixels. The inverse transformation is possible via [1]

$$p^I = T_C^I(p^C; g) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \left[p^C \frac{g_f}{p_z^C} + \begin{bmatrix} g_x \\ g_y \\ g_f \end{bmatrix} \right] \tag{21}$$

Where p_z^C is the z component of p^C and g_f is the focal length.

Transformation from Camera to Head.

The camera head is mounted at an angle with respect to the aircraft body. This angle is given by a set of roll, pitch, and yaw Euler angles and is contained as a subset of the camera parameters g.

$$g_A = \begin{bmatrix} g_\phi \\ g_\theta \\ g_\psi \end{bmatrix} \tag{22}$$

g_ϕ is a rotation about the x-axis (roll), g_θ is a rotation about the y-axis (pitch), and g_ψ is a ration about the z-axis(yaw). The transformation T_C^H is a cascade of three rotations in the following order: yaw, pitch, roll. Each rotation is done via a direction cosine matrix then combined to form a single rotation. In succession each rotation is defined as

$$C_\psi = \begin{bmatrix} \cos(\psi) & \sin(\psi) & 0 \\ -\sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$C_\theta = \begin{bmatrix} \cos(\theta) & 0 & -\sin(\theta) \\ 0 & 1 & 0 \\ \sin(\theta) & 0 & \cos(\theta) \end{bmatrix}$$

$$C_\phi = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\phi) & \sin(\phi) \\ 0 & -\sin(\phi) & \cos(\phi) \end{bmatrix}$$

with a general rotation from a navigation coordinate system to a body coordinate system given as $C_N^B = \boldsymbol{C_\phi C_\theta C_\psi}$. With this, any vector can be transformed from a navigation frame to a body frame from using $\boldsymbol{v}^B = C_N^B \boldsymbol{v}^N$ and in reverse using $\boldsymbol{v}^N = C_N^{B^T} \boldsymbol{v}^B$ [**1**].

With this the Camera to Head transformation is done via two rotations. First, a rotation around the camera frame z-axis is done to align the $\boldsymbol{x}$ and $\boldsymbol{y}$ axis of the camera frame to the camera head $\boldsymbol{x}$ and $\boldsymbol{y}$ axis. The rotation was not included in [**1**] but is required in order to use the rotation cascade above. The final rotation from camera to head is then

$$\boldsymbol{p}^H = C_{C_i}^H C_C^{C_i} \boldsymbol{p}^C \tag{23}$$

Where $C_C^{C_i}$ is a rotation of $[0, 0, \frac{3\pi}{2}]$ and C_H^C is the cascade of Euler angles contained in $\boldsymbol{g}_A$.

Transformation from Head to World.

The rotation from Head to World is similar as that of the Camera to Head transformation. Rotation angles of the aircraft are provided by an IMU with respect to the navigation frame and are time stamped to each image. These angles are provided in reference to the IMU world coordinate system of North, East, and Down, while the world coordinate system axes are East, North, and Up. A similar cascade of rotation angles as in the camera to head transformation is performed using the Euler angles provided by the IMU $\boldsymbol{s}_A = [\boldsymbol{s}_\phi \quad \boldsymbol{s}_\theta \quad \boldsymbol{s}_\psi]^T$ with an additional conversion from the navigation to world coordinate system.

$$C_W^N = C_N^W = C_0 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & -1 \end{bmatrix}$$

The transformation from Head to World is then given by

$$\boldsymbol{p}^W = C_0 C_H^N \boldsymbol{p}^H \tag{24}$$

Where $C_N^H = \boldsymbol{C}_\phi \boldsymbol{C}_\theta \boldsymbol{C}_\psi$ and $C_H^N = C_N^{H^T}$.

In order to determine the final ground projection the vector $\boldsymbol{p}^W$ must be appropriately scaled. The scaling is done by extending a vector from the camera position provided by the IMU $\boldsymbol{s}_W$ along $\boldsymbol{p}^W$ until it intersects the ground. The camera position is first corrected for the translation $\boldsymbol{g}_W$ to obtain $\boldsymbol{p}^W$

$$\boldsymbol{p}^W = \boldsymbol{s}_W + \boldsymbol{g}_W \tag{25}$$

$\boldsymbol{p}_W$ is then scaled so that it extends to the ground.

$$\boldsymbol{l}^W = \boldsymbol{p}^W \left| \frac{h_2 - h_1}{\boldsymbol{p}_U^W} \right| \tag{26}$$

Figure 6 illustrates the variables h_1 and h_2. In this h_2 is the height of the camera above the WGS84 ellipsoid and is contained in $\boldsymbol{s}_W$, and h_1 is the height of the projection plane above the WGS84 ellipsoid. The final location of the projection is given by

$$\boldsymbol{e}^W = \boldsymbol{p}^W + \boldsymbol{l}^W \tag{27}$$
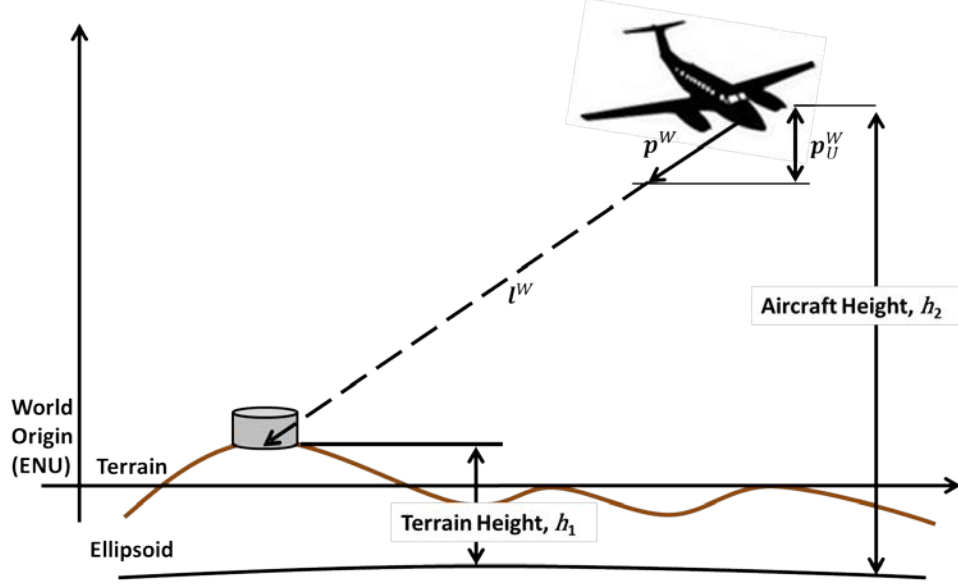
Figure 6. Vector Projections [1]

Radial Distortion.

The original projection algorithm presented in [1] does not include an adjustment for lens radial distortion. A goal of this work is to allow for the use of a variety of lower cost camera and lens combinations, the lens distortion parameters have been added to the projection. In this work, the distortion is modeled using a second order model. This selection was made to allow for comparison with the baseline calibration software [9] [15]. The distortion coefficients are given as $g_\rho = [g_{\rho 1} \quad g_{\rho 2} \quad g_{\rho 3}]^T$, with $g_{\rho 3} = 0$. This allows for future work and expansion of the algorithm if needed.

The final transformation from world to the image frame is to distort the pixel according to

$$\hat{p}_x^I = p_x^I + p_x^I \left[g \left({p_x^I}^2 + {p_y^I}^2 \right) + g_{\rho 2} \left({p_x^I}^2 + {p_y^I}^2 \right)^2 \right] \tag{28}$$

$$\hat{p}_y^I = p_y^I + p_y^I \left[g_{\rho 1} \left({p_x^I}^2 + {p_y^I}^2 \right) + g_{\rho 2} \left({p_x^I}^2 + {p_y^I}^2 \right)^2 \right] \tag{29}$$

27

Where $(\boldsymbol{p}_x^I, \boldsymbol{p}_y^I)$ and $(\widehat{\boldsymbol{p}}_x^I, \widehat{\boldsymbol{p}}_y^I)$ are the undistorted and distorted pixel locations respectively. For the reverse operation, from image to world frame, the location of the undistorted pixels is calculated via a linear least-squares routine to solve the system of equations [16].

This projection is central to the Bundle Adjustment calibration methodology presented in chapter 2 and serves as the basis for the initial 3D point estimates as well all the 2D measurement estimates. The next section will discuss a method for calibrating an aircraft mounted camera using feature detection and bundle adjustment.

Online Camera Calibration

A combination of precise navigation, image feature detection, and bundle adjustment are used to determine the calibration parameters of an aircraft mounted camera. No prior knowledge of the scenes contained in the images is required; however, knowledge of the terrain height will aid in the final solution. This section provides an overview of an algorithm originally developed by Cohenor and van Grass [5] with modification as noted.

Feature Detection and Correlation.

A central part in the projection algorithm is the determination of the pixel locations of a feature and the ability to correlate them across sets of images. This allows for the formation of the measurement vector $\boldsymbol{m}$. In this algorithm, feature detection is done via the Speeded Up Robust Features algorithm (SURF) [17]. The in-depth methodology and characteristics of feature detection are outside the scope of this report and will be omitted for conciseness. The SURF algorithm uses a Fast-Hessian detector to automatically detect the features in an image. There are three parameters that are varied in the detector:

a Hessian threshold, a sample step, and a parameter that controls the number of octaves in a Gaussian pyramid. The combination of all three parameters determines the size and number of features detected. Each feature found by SURF is given in (x, y) pixel coordinates is assigned a descriptor that allows for an initial correlation in successive images. The feature sets in each image are time stamped with the appropriate epoch and compiled. Initial feature correlation is done by comparing each descriptor in successive images to find the closest statistical match. This set of initial matching features is then processed using RANSAC that calculates a homography between the images. This ensures there are no outliers or incorrect matches. Next a unique feature number is assigned to each feature across successive images. This allows for the sorting and calculation of how many images the feature was visible. The initial measurement vector for the bundle adjustment is the compiled in the form $\boldsymbol{p}^I = (\boldsymbol{p}_{11} \dots \boldsymbol{p}_{1m}, \boldsymbol{p}_{21} \dots \boldsymbol{p}_{2m,} \boldsymbol{p}_{n1} \dots \boldsymbol{p}_{nm})$ where $\boldsymbol{p}_{11}$ is a 1×2 vector of feature one in image one, $\boldsymbol{p}_{21}$ is feature one in image two, etc. This is shown graphically in Figure 7. With the measurement vector in place the next step is to calculate the initial 3D points that are used to build $\boldsymbol{m}_0$.
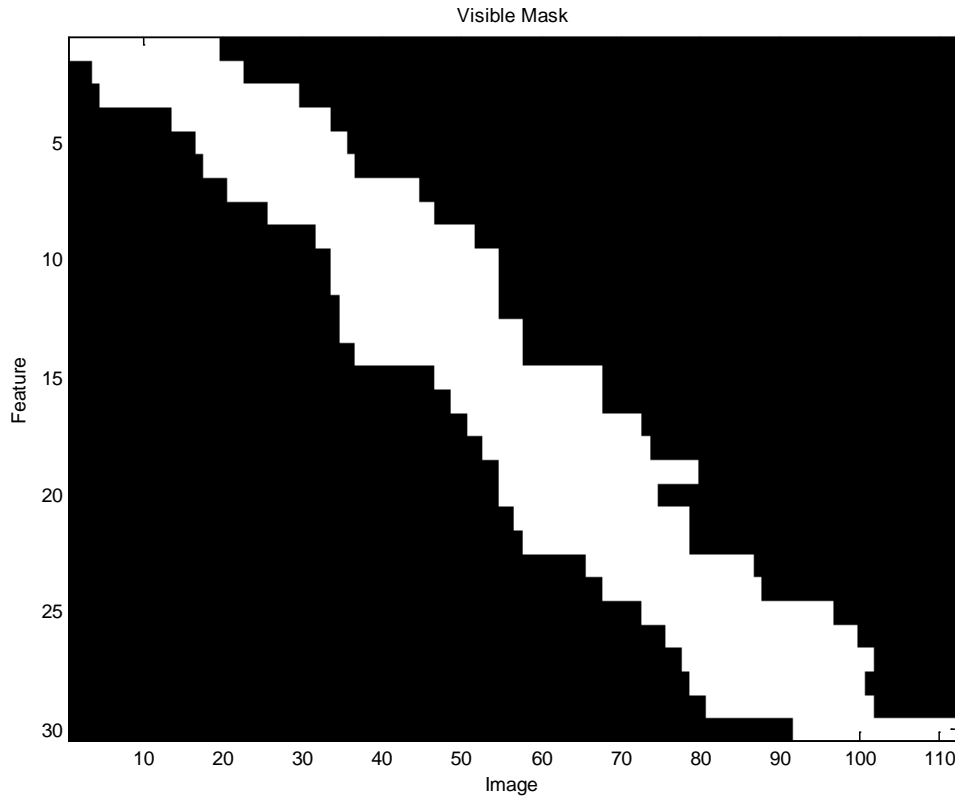
Figure 7. Visibility Mask. Visibility mask for 30 detected features over 120 images. Areas in white indicated that the feature was visible.

Initialization.

The LM algorithm requires an initial parameter vector on which it can iterate. The initial camera parameters are input as the best known calibration parameters, either from a specification sheet, bore site, or ground calibration. If an initial parameter value is unknown the algorithm will accept zero as initial value. The initial set of 3D points are calculated using the measurement vector p^I, aircraft position and attitude data s, and DTED using projection model previously presented. Since the projection is done with no prior knowledge of the scene h_1 is set to the DTED value of the aircraft position coordinates. Since each sequence of features should map to a single ground point, an average is taken of the East and North components of each feature set. This East and

North average position is compared against a set of DTED and the Up component is replaced with the elevation data. This approach provides an initial best guess at the world position without the need for iteration. The initial set of camera parameters are compiled into a single vector of the form $\boldsymbol{m} = (\boldsymbol{g}, \boldsymbol{s}_1 \dots \boldsymbol{s}_n)$, where $\boldsymbol{g}$ is the 1×12 vector of camera parameters and each $\boldsymbol{s}$ is the $1 \times 3j$ vector of the set of 3D coordinates corresponding to a single feature set, and n designates the number of feature sets. These parameters combined with the initial measurement vector can now be iterated using the LM algorithm.

Levenberg-Marquardt.

The initial step in the LM algorithm is to compute the initial measurement estimate $\widehat{\boldsymbol{p}}^I$ and the Jacobian matrix $\boldsymbol{J} = \partial \widehat{\boldsymbol{p}}^I / \partial \mathbf{m}$. The Jacobian is calculated numerically by incrementing each parameter to $\mathbf{m}_n + \delta$, then projecting the new parameter from world to image and taking the ratio with respect to δ. One can see that due to the nature of the partitioning of the parameters the derivative of any feature set that does not have a corresponding set of 3D coordinates in $\mathbf{m}$ will be zero. This fact results in a very sparse matrix as shown in Figure 8. The first twelve columns are the projections with linearized camera parameters; the remaining blocks are the projections with linearized coordinates. The sparse nature of this matrix can allow for some computational savings and is addressed in [**11**] and [**14**]. With the Jacobian in place, $\epsilon = \boldsymbol{p}^I - \widehat{\boldsymbol{p}}^I$ and $\boldsymbol{J}^T \boldsymbol{J}$ are calculated as required by the augmented normal equations. The matrix $\boldsymbol{J}^T \boldsymbol{J}$ also has a very sparse structure that can be exploited; one example is shown in Figure 9. The algorithm proceeds as in Figure 4 until one of the stop conditions are met. Lastly, the

31

calibration parameters and estimated world coordinates are extracted from the final $\boldsymbol{m}$.

The covariance of each parameter is calculated from $\boldsymbol{J^T J}$ and is discussed in the next section.

Covariance.

As previously stated, the sparse nature of the $\boldsymbol{J^T J}$ matrix can be exploited. The matrix is segmented into four distinct blocks as shown in Equation (30) and Figure 9.

$$\boldsymbol{J^T J} = \begin{bmatrix} \boldsymbol{U} & \boldsymbol{W} \\ \boldsymbol{W^T} & \boldsymbol{V} \end{bmatrix} \tag{30}$$

The covariance of the camera parameters is calculated using [11]

$$\Sigma_a = (\boldsymbol{U} - \sum_i \boldsymbol{W_i V_i W_i^T})^+ \tag{31}$$

Where the subscript i indicates each 3×3 block of V_i along the diagonal and the corresponding 12×3 block of W_i and the superscript $+$ indicates the pseudo-inverse. The derivation of the covariance extraction of the $\boldsymbol{J^T J}$ matrix is contained in Appendix 6 of [11].
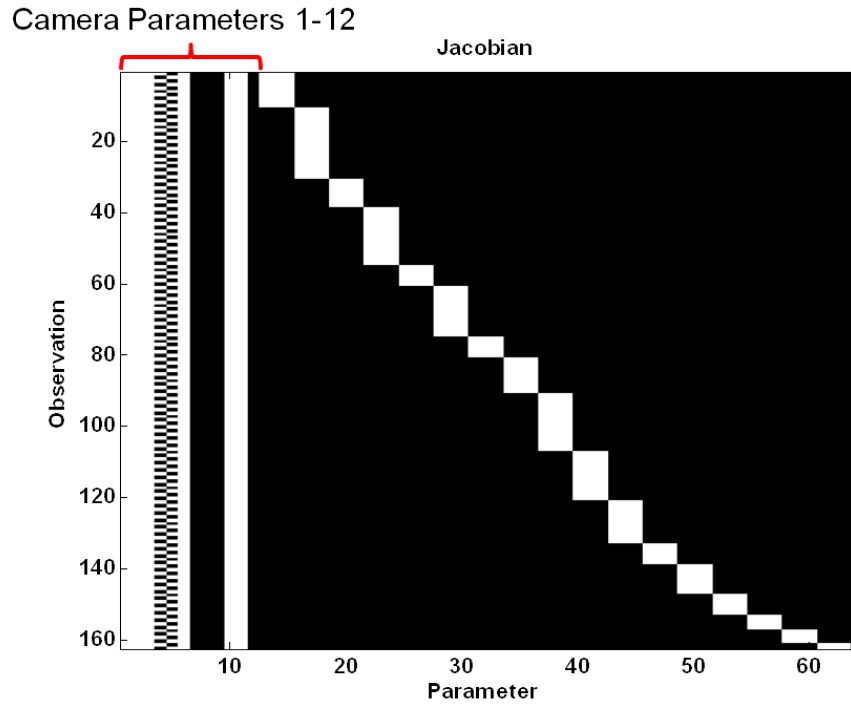
Figure 8. Structure of the Jacobian Matrix J. Black areas are populated with zeros.
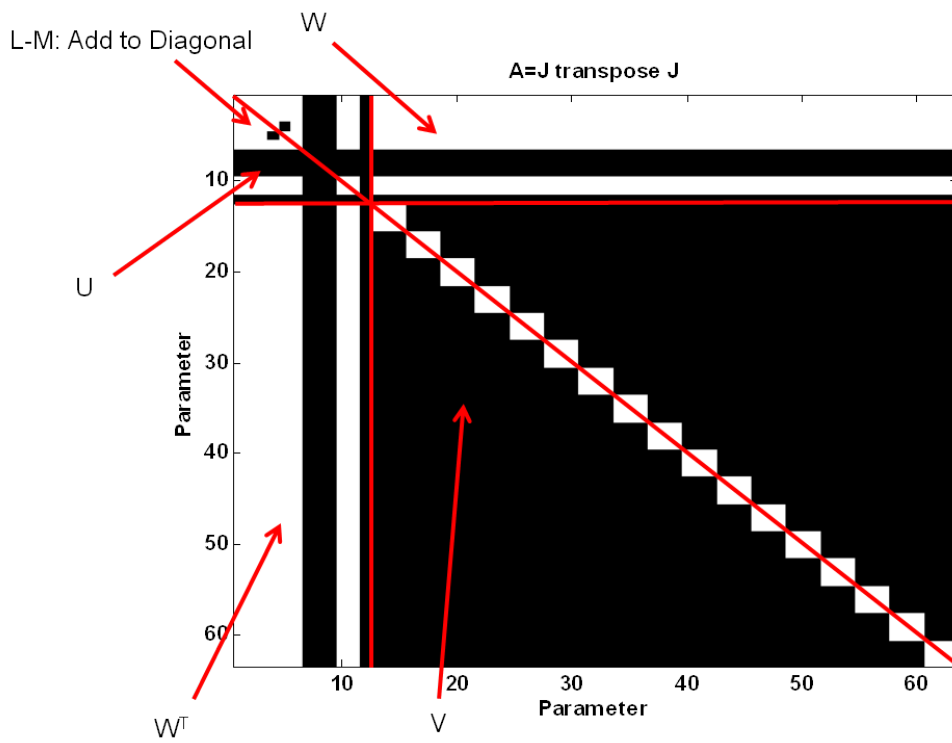
Figure 9. Structure of the $J^T J$ matrix. Black areas are populated with zeros.

Trajectory and Imaging Simulation.

Coehnor and van Grass [**5**] have performed only limited testing and analysis on the viability of the automated calibration routine. No analysis has been done on the factors that affect attaining accurate calibration results from the automated algorithm. This section describes a software package developed that is capable of manipulating the calibration algorithm input variables with the goal of performing a comprehensive sensitivity analysis on the effects of varying aircraft maneuvers and camera lookdown angle. The software loads a set of constants, generates a feature set and aircraft trajectory, and projects those features onto a series of simulated images. The details of each routine are examined in detail in this section.

Parameters.

In order to maintain a consistent set of constants, initial conditions and variables a routine to load all such information is implemented. All information regarding the terrain model, trajectory model, true camera parameters, and initial camera parameters from which to iterate are contained in this routine. Once loaded the parameters are not altered and are called in each successive routine.

Feature Generation.

To accurately recreate the sets of features detected by SURF a realistic terrain model of the world is generated. An area of interest for the simulated aircraft to 'fly' through is defined by entering a set of latitude and longitude coordinates. The topography of the terrain is then modeled using the highest resolution of DTED available. In this case, in

order to coordinate with the flight test data, the area around Edwards AFB, CA is modeled using DTED accurate to 1 arcsecond downloaded from the U.S. Geological Survey [18]. In order to simulate a feature as would be detected with SURF, the world is populated with a set of point coordinates. Two methods are implemented, the features can be distributed throughout the set coordinates in either a random distribution or in an equally spaced grid. The number of features can be set independently of the size of the simulated world in order to increase or decrease the density of features in each image. The random distribution was used at the baseline for the results presented in chapter four. The gridded feature layout is useful in controlling the number of features per image by shifting the origin of the aircraft trajectory. The elevations of the features are looked up in the DTED that is loaded to form the world topography. These features are assigned a simulated SURF descriptor and feature number to allow for sorting and correlation in subsequent routines. Because the descriptor and number are assigned to the feature as generated in the 'world' setting and not the image, they are persistent throughout the simulation and allow for exact feature matching across images. In order to simulate the effects of an urban location or incorrect DTED in which the elevation of the terrain may not be correct in DTED, an option is included to add an elevation to the features. Two options are available, either to add a set elevation to each feature or to implement a random distribution of elevations across the series of features. With the features assembled, the software can move on to trajectory generation.

Aircraft Trajectory Generation.

In order to simulate the aircraft maneuvers a trajectory generation software package is used. ProfGen is an aircraft trajectory generator developed by the Air Force Research Laboratory, Sensors Directorate [**19**] and allows for simulation of an aircraft flight path. ProfGen allows the user to build a complete aircraft trajectory by loading performance data about the aircraft and then implementing several baseline maneuvers. Straight and level flight, climbs, descents, and turns are cast into the software and put together into a final aircraft trajectory. Fourteen different trajectories were built using predicted performance data from the flight test aircraft (C-12C) and are called with each simulation run. The 14 trajectories implemented are:

1. 360° Degree Turn – A 30° bank turn in to the direction of the camera through a full 360° of azimuth.
2. 180° Degree Turn – A 30° bank turn in to the direction of the camera through a 180° of azimuth.
3. 90° Degree Turn – A 30° bank turn in to the direction of the camera through 90° of azimuth.
4. 30° Degree Turn – A 30° bank turn in to the direction of the camera through 30° of azimuth.
5. 15° Degree Turn – A 30° bank turn in to the direction of the camera through 15° of azimuth.
6. 360° Climbing Turn – Executed the same as above except with an 11° pitch up.
7. Holding Pattern – A standard holding pattern with 30 second straight legs and 180° turns using 30° angle of bank at each end.
8. S-turn – Two 30° bank turns with a reversal after 90° of heading change.
9. Straight – A straight and level course for 30 seconds.
10. Straight Climb – A straight course with an 11° pitch up for 30 seconds.
11. 1° Bank Turn – A 1° bank turn in to the direction of the camera through 30° of azimuth.
12. 5° Bank Turn – A 5° bank turn in to the direction of the camera through 30° of azimuth.

13. 10° Bank Turn – A 10° bank turn in to the direction of the camera through 30° of azimuth.
14. 15° Bank Turn – A 15° bank turn in to the direction of the camera through 30° of azimuth.

Each run was done at a simulated altitude of 3000 meters MSL with an airspeed of 90 meters per second. The ProfGen configuration files, as run, are included in Appendix B.

The simulation requires the input of at least an aircraft position and attitude at specific time intervals. ProfGen provides an output file of the aircraft state at the time step specified by the user, in this case four Hertz. The aircraft position and attitude data along with the time at each step are extracted from the output log and are assembled into a single file to allow for the time correlation of the simulated images. If desired, the navigation solution can be injected with noise. The measurement noise is from a normal distribution with a user assignable standard deviation. With the 'world' populated and a trajectory generated the ground features are projected on to images as described in the next section.

Image Generation.

Image generation is done using the projection presented in the Pinhole Projection Model in reverse. The ground features are projected from the world frame to the image frame. For each time step of the aircraft trajectory the collection of features is projected into the image frame. Features outside of the set image size ($1600 \, x \, 1200$ in this case) are discarded and the resulting 'image' is stored along with the associated descriptor and feature number. The user has the option to add radial distortion and/or pixel noise to the final image projection. The radial distortion is done as described in chapter 2 and allows for the input of the second order radial distortion coefficients. The pixel distortion

37

simulates the uncertainty of SURF to be able to detect a feature at the exact correct pixel in each image. Pixel noise is added to the final projection in the form of a random normal distribution with a mean of one and a standard deviation set by the user. Finally, a single compendium of data is assembled that contains each time step, all features in that time step, the image frame and world frame location of the features, and the aircraft position and attitude. The final output of the simulation can be displayed graphically in three separate ways in order to aid in debugging and visualization of the scene. A three dimensional view of the world is available that displays the aircraft and camera field of view on the terrain. To simulate the field of view of a camera, four additional projections are done. These projections are from the image to world frame and are at the four corners of the image. A top down view is available which more easily depicts where the camera is imaging. And finally a view of the simulated image is available to allow for the correlation of the two outside views. An example of each view is shown in Figure 10 through Figure 12.

The data collected from the simulation software is fed directly into the calibration algorithm along with the initial conditions contained in the parameters file. After the calibration routine is complete, the standard deviation of each evaluated parameter is calculated. Given that the simulation uses known features generated in the world frame an error calculation can be done by doing a projection using the resulting calibration parameters. A projection from image to world is done for each feature in the set of images. The projected locations are subtracted for the true locations of the features and a horizontal root mean square error is calculated.
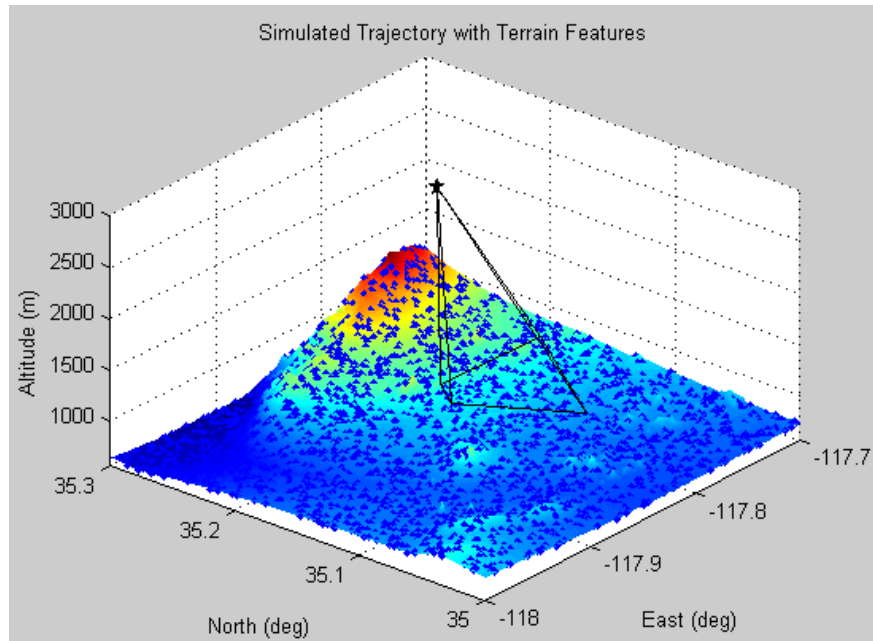
Figure 10. Simulated Trajectory. Solid lines projected onto the terrain indicate the camera field of view, and the blue dots on the terrain are the generated features.
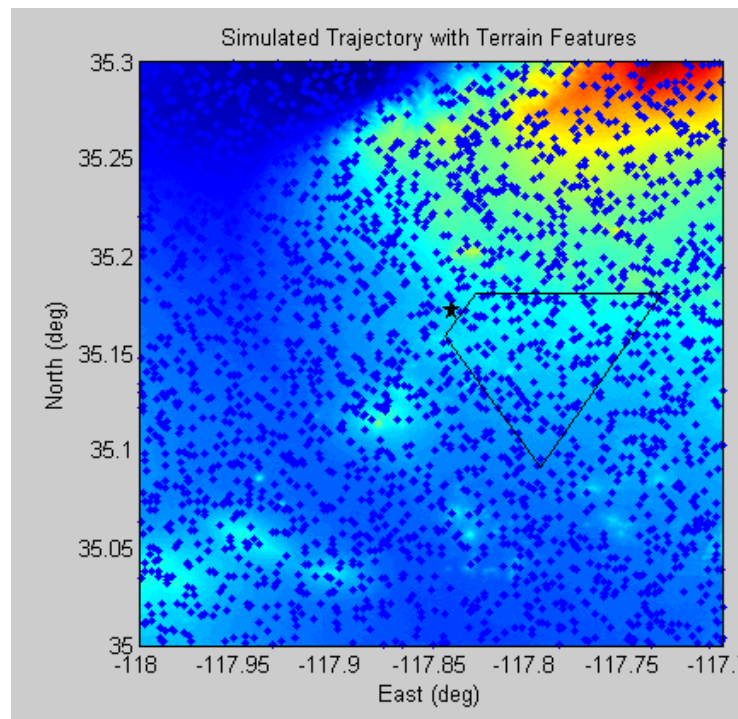
Figure 11. Overhead View. Solid lines depict the camera field of view, and the blue dots on the terrain are the generated features.
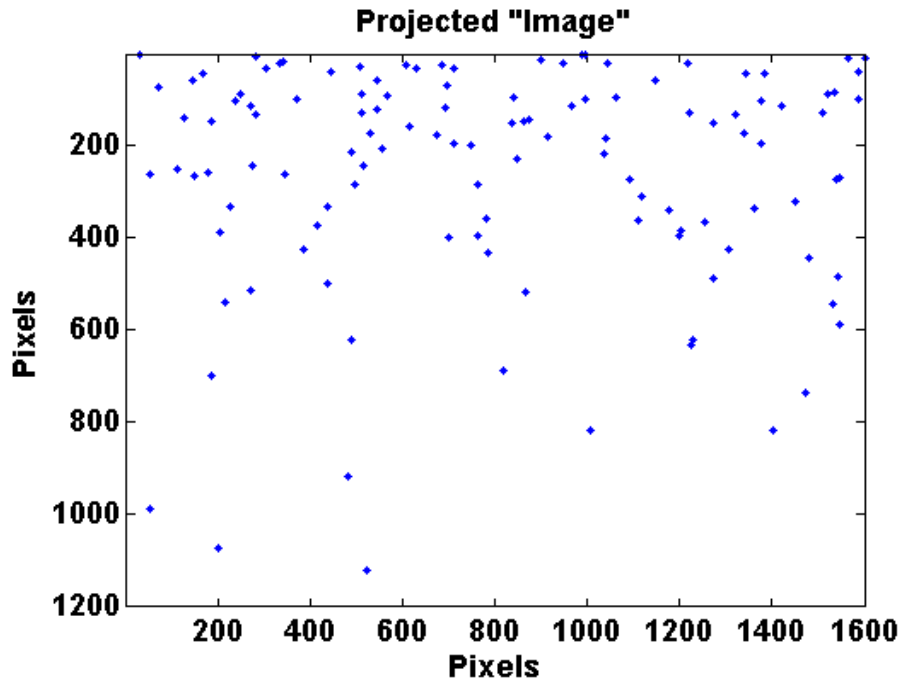
Figure 12. Simulated Image. The blue dots correspond to the image plane locations of the features observed in one time instance of the trajectory.

IV. Simulation and Flight Test Results

The section illustrates the application and results obtained from the concepts presented in chapter three. The camera calibration routine is implemented in three different scenarios to evaluate sensitivity to differing camera orientation and aircraft trajectories, measurement noise, radial distortion, and initial conditions, and combinations thereof. First, the simulation is used to evaluate the algorithm given several camera orientations and proposed aircraft trajectories, and a "best case" is determined. Second, experimental data collected during a flight test is used and compared to the simulation results. Last, the flight test collected aircraft trajectory data is used in conjunction with the simulation generated image sets to further validate the simulator.

Simulation

The simulation portion of the analysis had two goals: to perform a comprehensive sensitivity analysis of the calibration algorithm and to narrow the scope of the flight test. Initially several camera orientations and aircraft trajectory combinations are tested using the methodology presented in chapter three in order to provide a baseline as well as expose any errors due to observability. A baseline trajectory and orientation is chosen to analyze the effects of navigation system and feature detection noise. Finally, the maneuvers are evaluated with an operationally representative level of measurement and feature noise. The goal of this analysis is to determine the effects of the aircraft trajectory and camera orientation, feature detection measurement noise, and initial conditions on the calibration routine and provide a best case maneuver. In order to control the effects of

incorrect feature matching, the simulation provided a persistent set of features that could be exactly correlated across the given trajectory profile. Though the total number of features detected and matched during a single trajectory varied from 200 to 1000 depending on the number of features generated and trajectory the total number of features used during the iteration process was capped at 60 for processing efficiency.

In all cases the offset vector, g_P, was set to a constant and not estimate during these simulations. Initial analysis showed a tendency for the term to dominate the solution if the algorithm was allowed to estimate g_P along with all of the other terms. The LM algorithm would reach a local minimum using only the offset bias, which led to the other calibration terms to be drastically incorrect.

Aircraft Maneuver and Camera Orientation Simulation Results.

To determine the effects of camera look angle, camera head angles were varied from zero to 30 degrees in roll, zero to 45 degrees in pitch, and from zero to 90 degrees in azimuth. These angles were chosen as being representative of how an actual aircraft camera would be set up. For each trajectory, each parameter was varied independently, and in conjunction with the other parameters. For example, the pitch was varied independently, then in conjunction with each yaw angle. In each case, the feature and image generation was done with a constant set of true calibration parameters and features matched exactly across images. Preliminary analysis had shown the propensity of the routine to be quite exact, for this reason each orientation, except the straight and level was run only one time. The calibration routine was initialized with a set of operationally representative parameters for the camera perspective center (CPC) and camera rotation

angles, and zero for the distortion coefficients. The initial conditions used were three degrees off in each angle, 30 pixels off in the CPC and 50 pixels off in the focal length. The fourteen trajectories described in chapter thee were used for this analysis.

Table 1 shows a sample of the results for one camera configuration from each of the four satisfactory maneuvers. For all trajectories and camera orientations outside of the straight and level trajectory the solution converged to exactly correct (zero error) values for all eight iterated calibration parameters. Further, the covariance values for all eight parameters were significantly small and similar across the spectrum of each parameter with no outliers. An average of seven iterations of the LM algorithm was needed for these cases. This data shows that the algorithm is capable of calculating an exactly correct solution with only the straight and level case being unobservable.

Table 1. Selected Simulation Results.

Parameter	Turn				Climbing Turn			
	True	Estimated	Error	Std. Dev.	True	Estimated	Error	Std. Dev.
g_ϕ (deg)	0	0	0	0.026	0	0	0	0.009
g_θ (deg)	30	30	0	0.010	30	30	0	0.015
g_ψ (deg)	30	30	0	0.012	30	30	0	0.004
g_x (pixels)	800	800	0	0.30	800	800	0	0.26
g_y (pixels)	600	600	0	0.36	600	600	0	0.18
g_f (pixels)	-1100	-1100	0	0.23	-1100	-1100	0	0.14
g (none)	-0.2543	-0.2543	0	0.0006	-0.2543	-0.2543	0	0.0008
g (none)	0.01543	0.01543	0	0.0008	0.01543	0.01543	0	0.001
Parameter	Holding Pattern				S-Turn			
	True	Estimated	Error	Std. Dev.	True	Estimated	Error	Std. Dev.
g_ϕ (deg)	0	0	0	0.008	0	0	0	0.014
g_θ (deg)	30	30	0	0.008	30	30	0	0.011
g_ψ (deg)	30	30	0	0.004	30	30	0	0.003
g_x (pixels)	800	800	0	0.17	800	800	0	0.25
g_y (pixels)	600	600	0	0.18	600	600	0	0.28
g_f (pixels)	-1100	-1100	0	0.14	-1100	-1100	0	0.20
g (none)	-0.2543	-0.2543	0	0.0006	-0.2543	-0.2543	0	0.0008
g (none)	0.01543	0.01543	0	0.0007	0.01543	0.01543	0	0.001

Results for the straight and level trajectories showed that the algorithm had difficulty converging to a correct solution. Each camera orientation for the straight and level case was run five times. Figure 14 shows that camera pitch angle was most difficult for the algorithm to determine with an error of one degree at zero pitch and almost seven degrees at 45 degrees pitch. An interesting result is that as the yaw angle approached 90 degrees, the error in the estimated error decreased. This effect is even more apparent in the yaw angle estimation as shown in Figure 15. When the camera look yaw angle approaches ninety degrees there is almost no error is the estimate yaw parameter. Table 2 shows results for four of the 28 test cases simulated. As evident in Table 2, the CPC

coordinates showed accuracies of as much as a tenth of a pixel until the camera pitch angle grew to 45 degrees wherein the errors grew to as much as 75 pixels in g_x and 35 pixels in g_y. Focal length displayed a similar trend with errors increasing as the pitch and azimuth angle increased. Likewise, the two distortion parameters were accurate to the hundredths for camera pitch angles less than 45 degrees. In each of the straight trajectory scenarios the algorithm iterated the maximum of 100 times.

The results show that when presented with image features that track linearly through the image, illustrated in Figure 16, the LM algorithm would converge to a poor estimate of the calibration parameters.
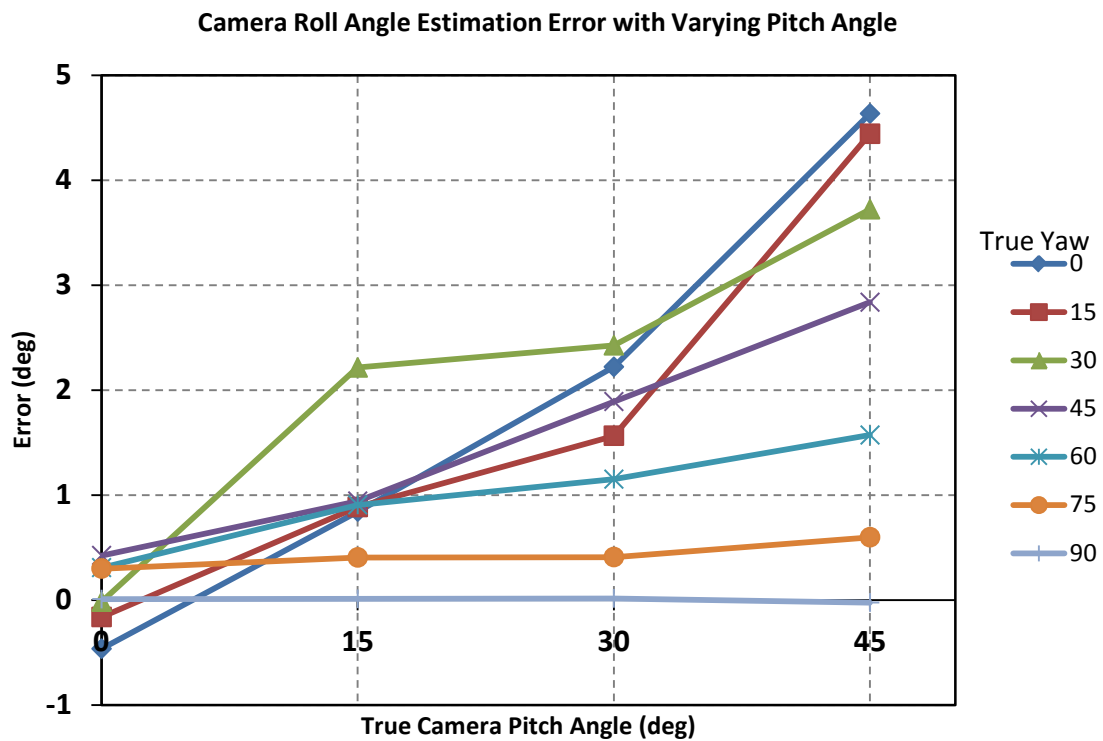
Figure 13. Camera Roll Angle Estimation Error, Straight Maneuver
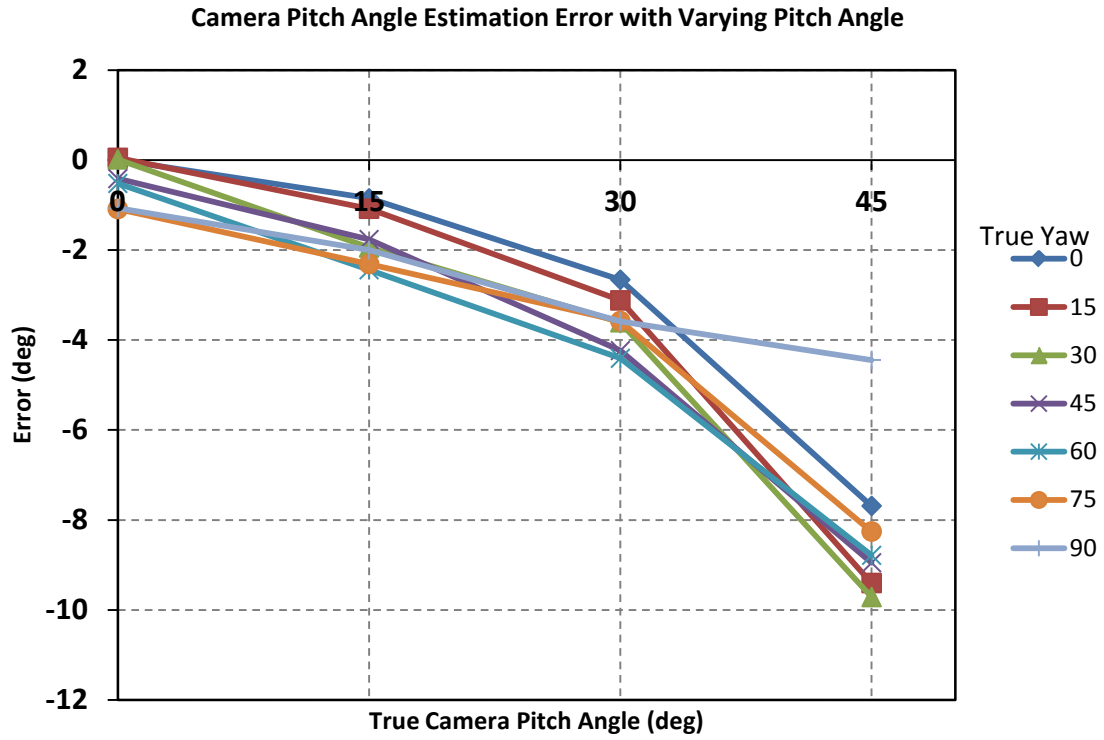
Figure 14. Camera Pitch Angle Estimation Error, Straight Maneuver
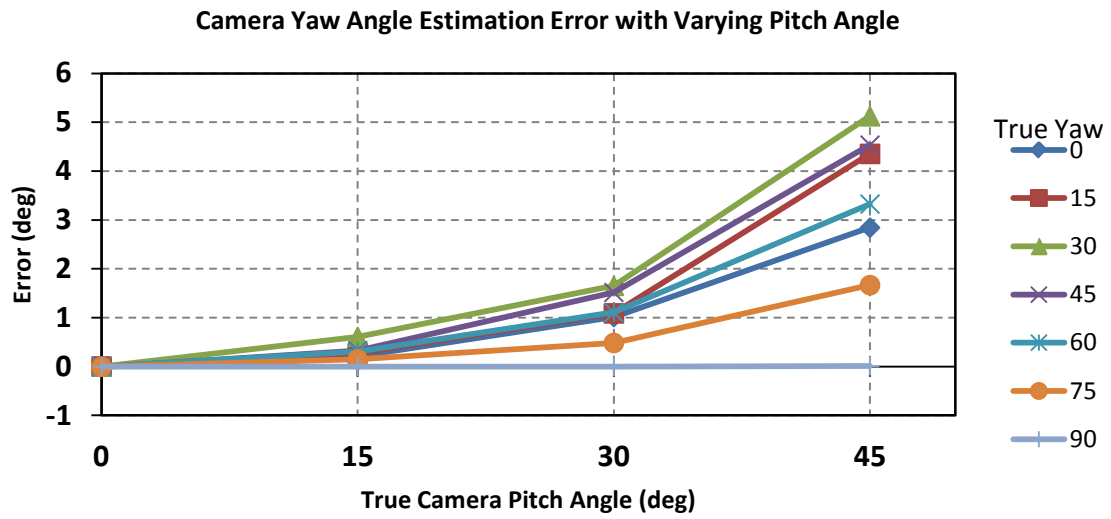
Figure 15. Camera Yaw Angle Estimation Error, Straight Maneuver

46

Table 2. Straight Trajectory Simulation Results

Parameter	Straight Case 1				Straight Case 2			
	True	Estimated	Error	Std. Dev.	True	Estimated	Error	Std. Dev.
g_ϕ (deg)	0	0.9	0.9	0.20	0	2.6	2.6	72.09
g_θ (deg)	0	0	0	0.02	15	13.5	-1.5	19.50
g_ψ (deg)	0	0	0	0.00	15	15.7	0.7	17.03
g_x (pixels)	800	800.0	0	11.55	800	799.9	0.1	15.62
g_y (pixels)	600	600.0	0	15.17	600	600.1	0.1	14.59
g_f (pixels)	-1100	-1070.3	29.7	10.86	-1100	-1041.8	58.2	0.01
$g_{\rho 1}$ (none)	-0.2543	-0.024	0.2303	0.001	-0.2543	-0.023	0.2313	0.001
$g_{\rho 2}$ (none)	0.01543	0.014	0.00143	0.001	0.01543	0.012	0.00343	0.001
Parameter	Straight Case 3				Straight Case 4			
	True	Estimated	Error	Std. Dev.	True	Estimated	Error	Std. Dev.
g_ϕ (deg)	0	3.5	3.5	59.74	0	3.2	3.2	62.13
g_θ (deg)	30	26.3	3.7	33.32	45	36.8	8.2	57.70
g_ψ (deg)	30	32.0	2	26.65	45	49.4	4.4	37.46
g_x (pixels)	800	799.9	0.1	11.80	800	847.1	47.1	17.50
g_y (pixels)	600	600.1	0.1	10.87	600	635.7	35.7	17.73
g_f (pixels)	-1100	-1022.3	77.7	0.01	-1100	-941.0	159	0.06
$g_{\rho 1}$ (none)	-0.2543	-0.022	0.2323	0.001	-0.2543	-0.017	0.2373	0.001
$g_{\rho 2}$ (none)	0.01543	0.012	0.00343	0.001	0.01543	0.007	0.00843	0.001
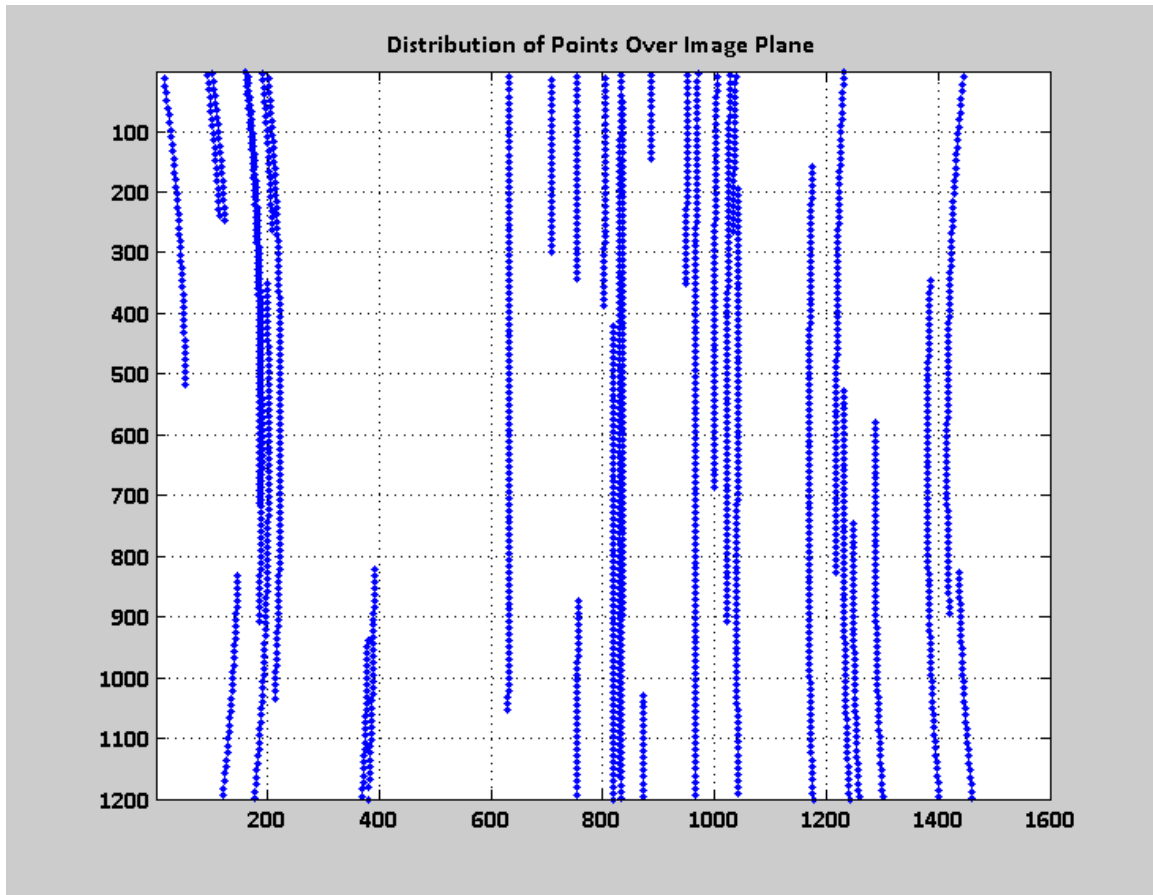
Figure 16. Composite Image. Blue dots indicate features as they track through the simulated image. In this instance, the features track from the top to the bottom of the image frame. Induced radial distortion is seen in the curving lines at the far left and right features.

Given that the straight trajectories were the only cases that did not results in accurate calibration estimates, several more incremental test cases were run in order to determine a minimum bank angle required for accurate parameter estimates. The results from Table 1 show that having a bank during the maneuver leads to satisfactory results. With this, the four camera orientations in Table 2 were run using simulation maneuvers 11 through 14 (chapter 3). It was found that at a zero degree look down (Table 2, Case 1) a bank angle of one degree produced accurate results, while the worst case (Table 2, Case 4) required a bank angle of five degrees for accurate results.

In order to determine if the amount of heading change had any effect on accurate results four more test cases were run. Simulation trajectories one through five were run and are shown in Table 3. All of these cases produced accurate results with zero error once again. This combined with the bank angle test case results show that having a bank angle on the aircraft of five degrees or more is desirable in attaining accurate parameter estimates.

Table 3. Heading Change Test Cases

Parameter	Case 1 - 15° Heading Change				Case 2 - 30° Heading Change			
	True	Estimated	Error	Std. Dev.	True	Estimated	Error	Std. Dev.
g_ϕ (deg)	0	0	0	0.087	0	0	0	0.027
g_θ (deg)	0	0	0	0.040	15	15	0	0.015
g_ψ (deg)	0	0	0	0.017	15	15	0	0.008
g_x (pixels)	800	800	0	1.091	800	800	0	0.405
g_y (pixels)	600	600	0	0.463	600	600	0	0.234
g_f (pixels)	-1100	-1000	0	0.414	-1100	-1100	0	0.276
$g_{\rho 1}$ (none)	-0.2543	-0.2543	0	0.002	-0.2543	-0.2543	0	0.001
$g_{\rho 2}$ (none)	0.01543	0.01543	0	0.002	0.01543	0.01543	0	0.001
Parameter	Case 3 - 90° Heading Change				Case 4 - 180° Heading Change			
	True	Estimated	Error	Std. Dev.	True	Estimated	Error	Std. Dev.
g_ϕ (deg)	0	0	0	0.037	0	0	0	0.031
g_θ (deg)	30	30	0	0.007	45	45	0	0.010
g_ψ (deg)	30	30	0	0.012	45	45	0	0.014
g_x (pixels)	800	800	0	0.213	800	800	0	0.239
g_y (pixels)	600	600	0	0.345	600	600	0	0.504
g_f (pixels)	-1100	-1100	0	0.297	-1100	-1100	0	0.251
$g_{\rho 1}$ (none)	-0.2543	-0.2543	0	0.001	-0.2543	-0.2543	0	0.001
$g_{\rho 2}$ (none)	0.01543	0.01543	0	0.001	0.01543	0.01543	0	0.001

Image Pixel Measurement Error Simulation Results.

As described in chapter three, the simulation is able to inject noise into the measured image pixel locations. Several cases were run to characterize the sole effects of noise in the measured feature locations. For all test runs a constant set of features and initial conditions were used, the only parameter varied was the standard deviation of the pixel noise. The turn maneuver with a look down angle of 45 degrees and yaw angle of 90 degrees was used for all cases. The projected image feature locations were injected with noise in the form of a Gaussian distribution with a mean of one and a set standard deviation in pixels. Each feature in each image was injected with a different amount of noise on the set interval. Average results from five runs each noise level are shown in Figure 17 and Figure 18. It can be seen that at noise levels above five pixels the estimation errors grow rapidly. At magnitudes above seven pixels the feature matching portion of the algorithm was not be able to correlate between the features. Even if the same world feature appeared across a series of images, it was given a different amount of image measurement noise at each instance which caused the algorithm to see them as different features. In this sense adding noise to the feature locations simulates miscorrelated features in the image set.
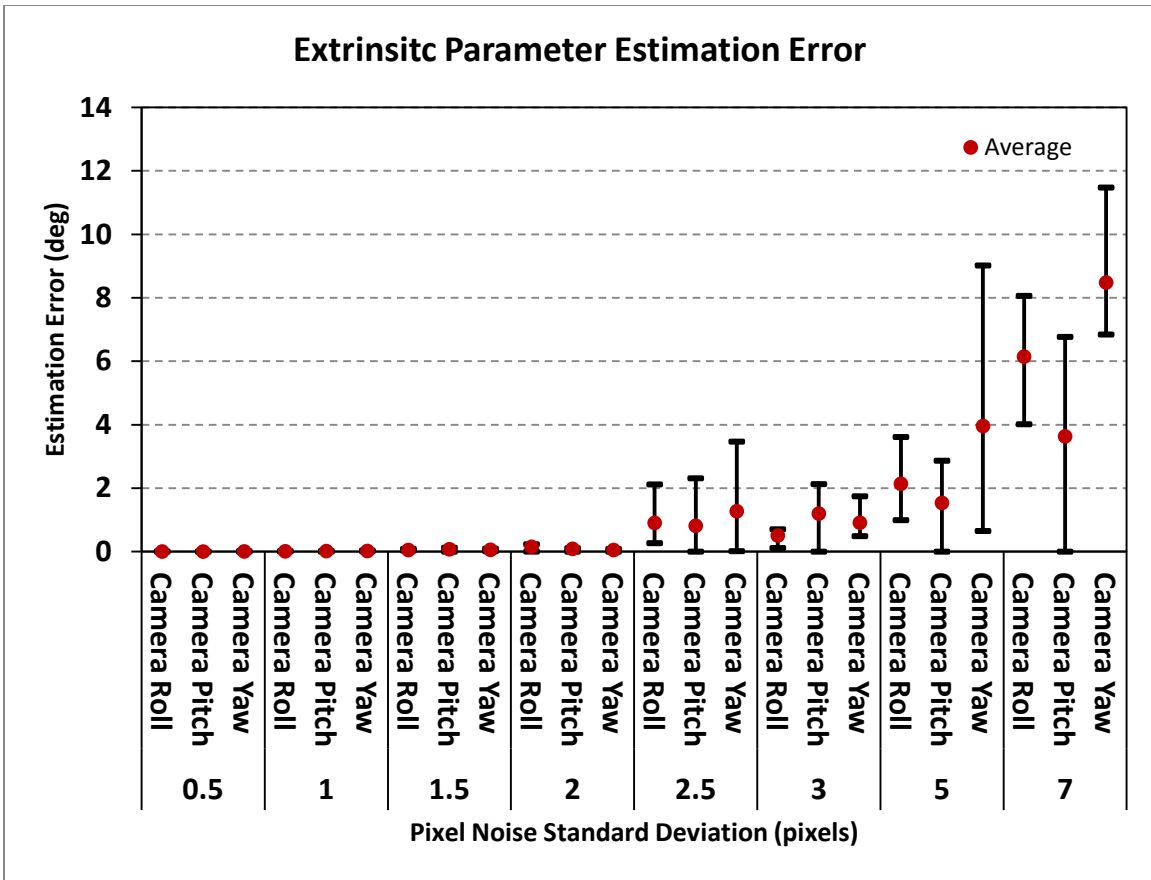
Figure 17. Extrinsic Parameter Estimation Error due to Feature Measurement Noise.
Error bars indicate the maximum and minimum error for the sample set.
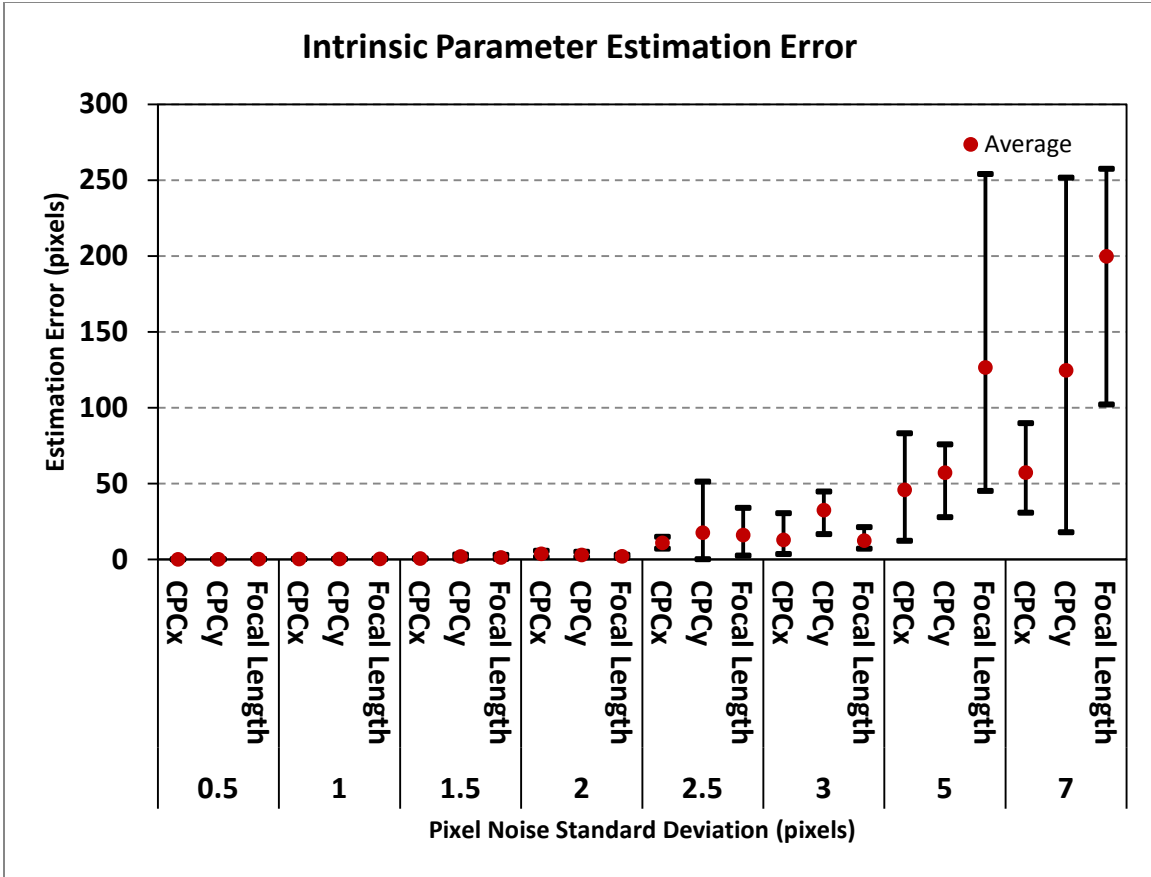
Figure 18. Intrinsic Parameter Estimation Error due to Feature Measurement Noise.
Error bars indicate the maximum and minimum error for the sample set.

Figure 19 shows the horizontal projection error of the noise injection on each interval. Though the horizontal root mean square (HRMS) projection error will change with differing maneuvers and camera orientations, it is useful in visualizing trends. It can be seen that errors of up to five pixels do not translate to very large projection errors. However, errors beyond five pixels can induce very large errors in the estimated parameters which leads to large projection errors.

Average Horizontal RMS Error
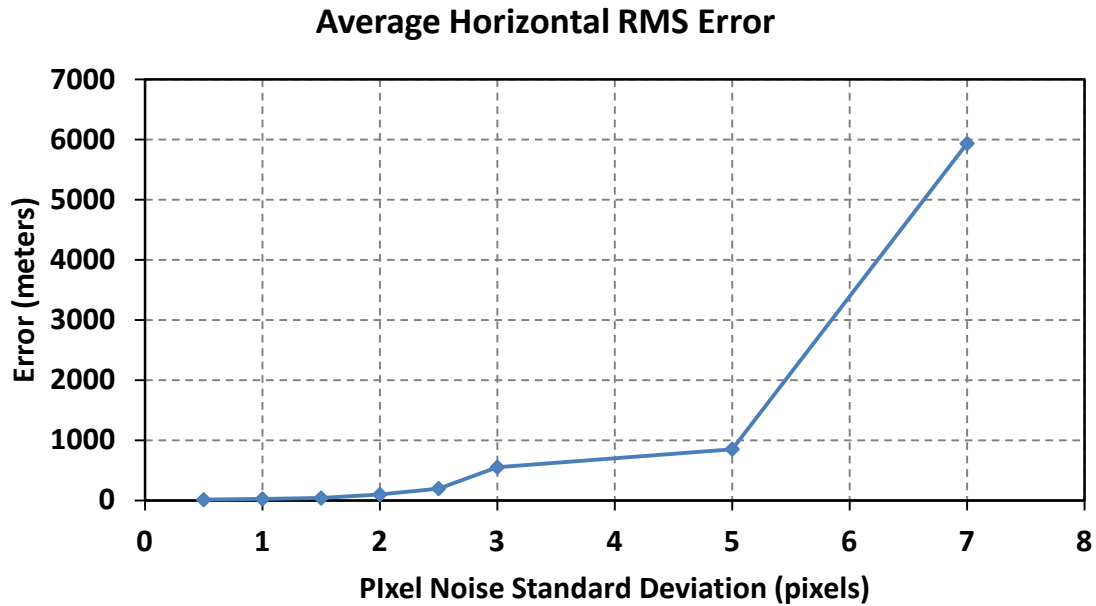
Figure 19. Average Horizontal RMS Projection Error Due to Feature Measurement Noise

Navigation Noise Simulation Results.

As described in chapter three, the simulation is able to inject noise into the navigation solution generated by ProfGen. To test the sensitivity of the algorithm to navigation position measurement noise several test cases were run. For all test runs a constant set of features and initial conditions were used, the only parameter varied was the magnitude of the navigation measurement noise. The camera orientation used in the feature noise section was used in these cases as well. The navigation position solution was injected with noise from a Gaussian distribution with a mean of one and standard deviations within the limits of the actual standard deviations seen in the flight test (Figure 32). Figure 20 and   shows the horizontal projection error at each noise level interval. Though the horizontal projection error will change with differing maneuvers and camera orientations, it is useful in visualizing trends. It can be seen that at noise levels

commensurate with that of the real navigation system solution only minor projection errors were evident.

Average Horizontal RMS Error
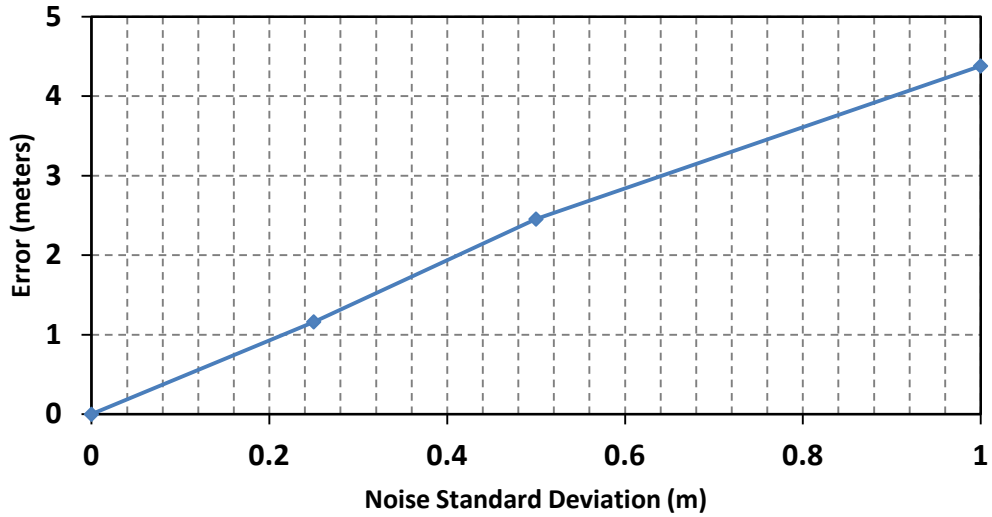
Figure 20. Horizontal RMS Error Due to Navigation Position Measurement Noise

Average Horizontal RMS Error
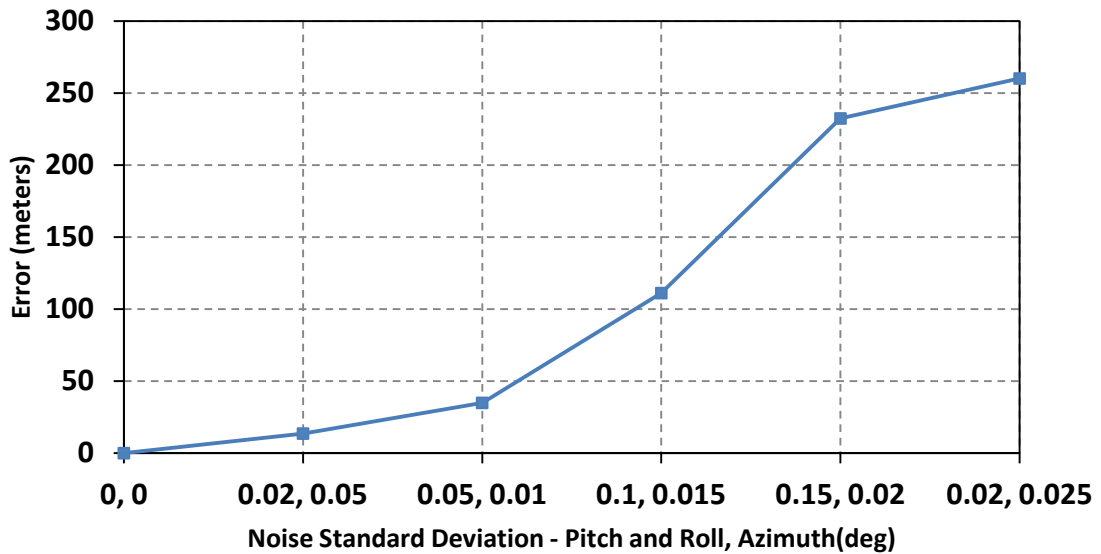
Figure 21. Horizontal RMS Error Due to Navigation Attitude Measurement Noise

Varying Initial Conditions Simulation Results.

A study of the effect of initial conditions was conducted using the best case trajectory. The camera pitch and yaw angle initial conditions were varied in three degree increments from 55 and 90 degrees to zero degrees. The initial CPC values were varied from 800 pixels in x and 600 pixels in y to zero in fifty pixel increments. The initial focal length was varied from the true value of -1000 pixels to zero in fifty pixel increments. No error in the final estimated parameters is seen until the initial angle conditions reached zero degrees in pitch and 36 degrees in yaw. The algorithm was able to correctly estimate the true CPC values even when initialed at zero. Initial focal length reached 600 pixels off from the true value before any error was seen.

Aircraft Maneuver and Camera Orientation with Noise Simulation Results.

To provide to a scenario in which a 'best case' combination of camera orientation and aircraft trajectory can be chosen, several more runs were done; 2,396 in all. Selected trajectories and camera orientations were run with a realistic set of navigation position and attitude noise and feature detection noise. The navigation position and attitude noise were set to values representative of the proposed flight test equipment. Standard deviations of 0.33 m, 0.18 degrees in pitch and roll and 0.5 degrees in were used [20]. Feature noise was induced with a two pixel standard deviation based on the pixel noise test presented earlier as well as information presented in [17]. The straight and level trajectory has been shown to produce poor estimation accuracies and was eliminated. The four remaining trajectories used were the turn, climbing turn, holding pattern and S-Turn. In order to narrow the scope of this analysis the camera orientations were limited

to those which would be representative of the flight test. Camera roll was set to zero degrees, pitch angle was varied from zero to 45 degree in fifteen degree increments and camera yaw was varied from zero to 90 degrees in forty five degree increments. To mimic a realistic flight, a new set of world features, measurement noises were used for each run. A two fold approach was used to determine a best case. First, the average errors in the camera angle estimation across all trajectories; then the error across the four maneuvers given the best case camera orientation. Figure 22 through Figure 27 illustrate the camera orientation parameter estimation error and standard deviation. Yaw angles of 90 and 45 degrees at 45 degrees pitch produced the lowest overall error. Similarly, standard deviations for both tend to be smallest at 45 degrees pitch. At 90 degrees of yaw, a pitch angle of 45 degrees is shown to have a slightly lower average standard deviation, and determined to be best. Angles of zero, 45, and zero have the largest errors and are a worst case.
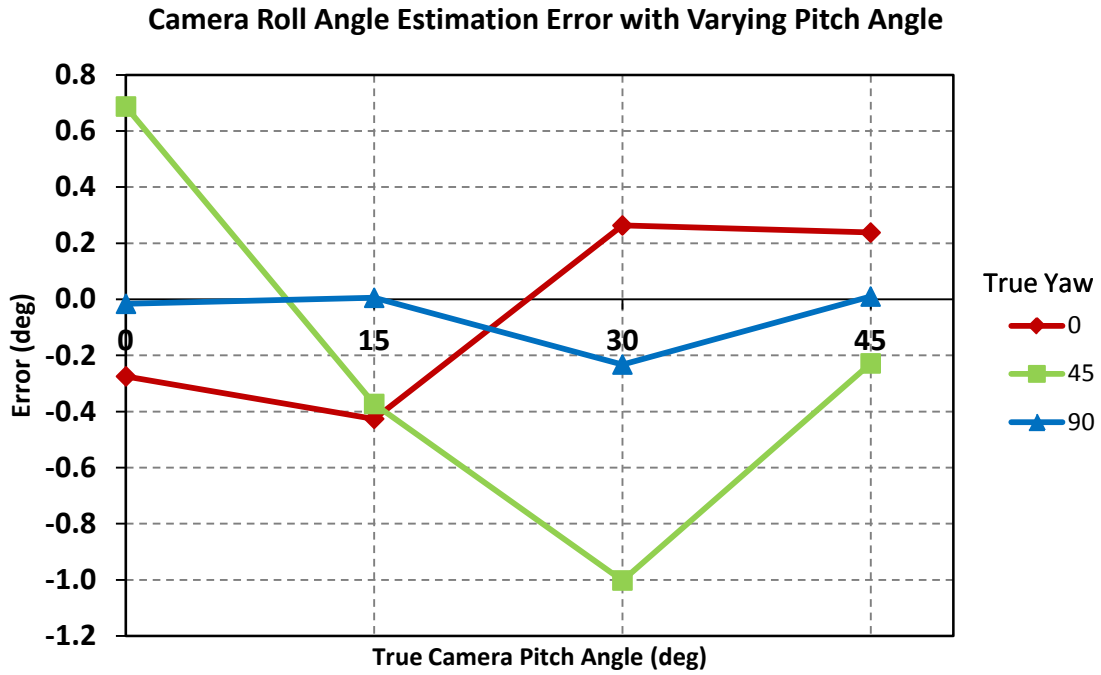
Figure 22. Camera roll angle average estimation error for all trajectories
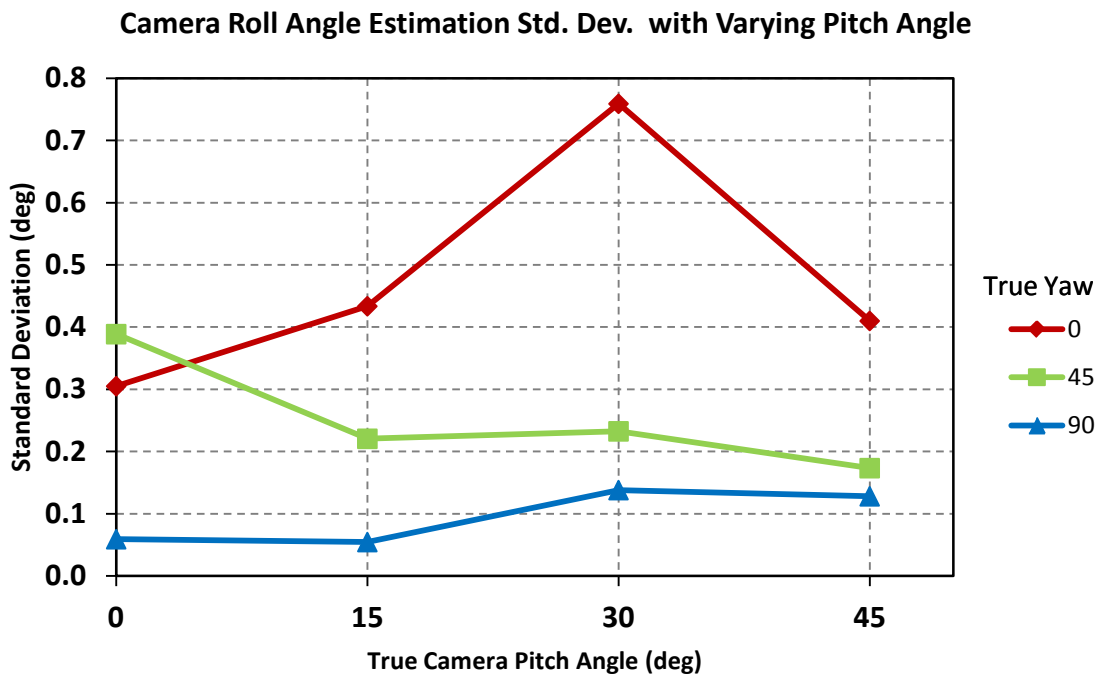
Figure 23. Camera roll angle estimation average standard deviation for all trajectories
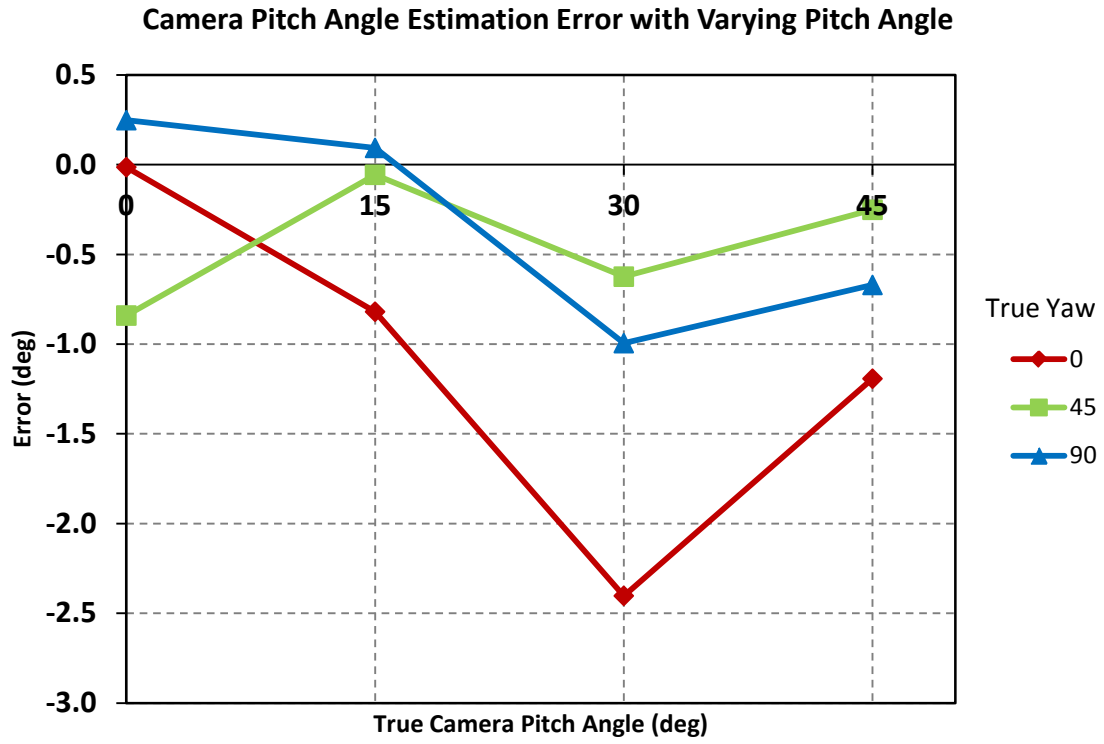
Figure 24. Camera pitch angle average estimation error across all trajectories.
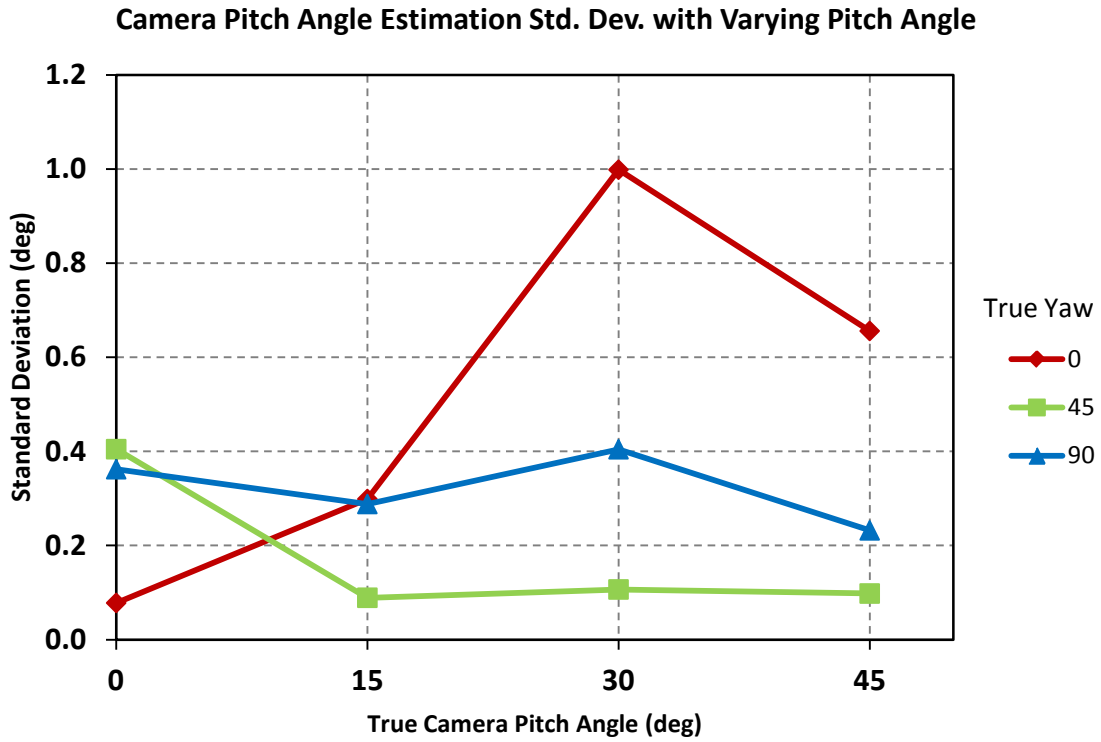
Figure 25. Camera pitch angle average estimation standard deviation across all trajectories.
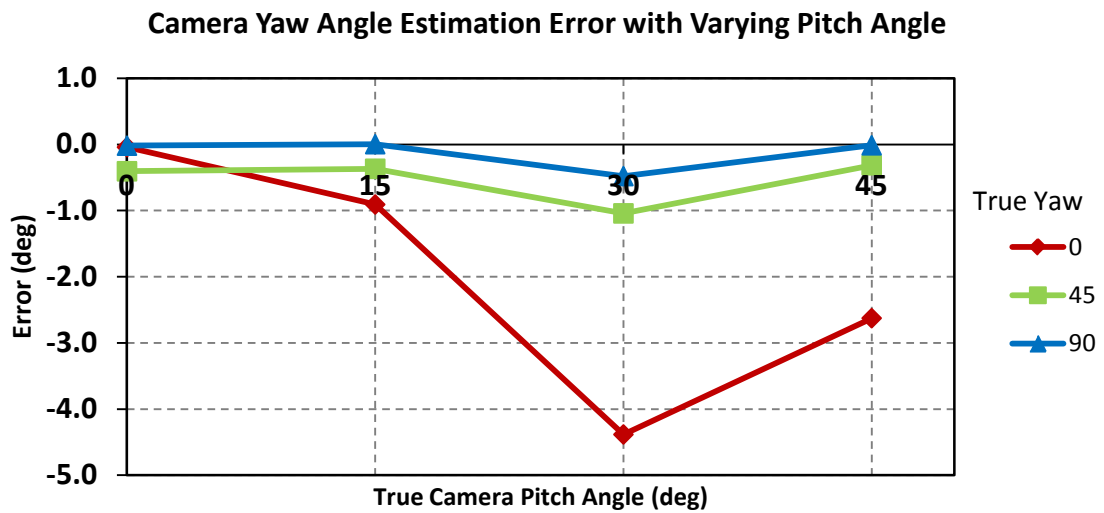
Figure 26. Camera yaw angle average estimation error across all trajectories.

Camera Yaw Angle Estimation Std. Dev. with Varying Pitch Angle
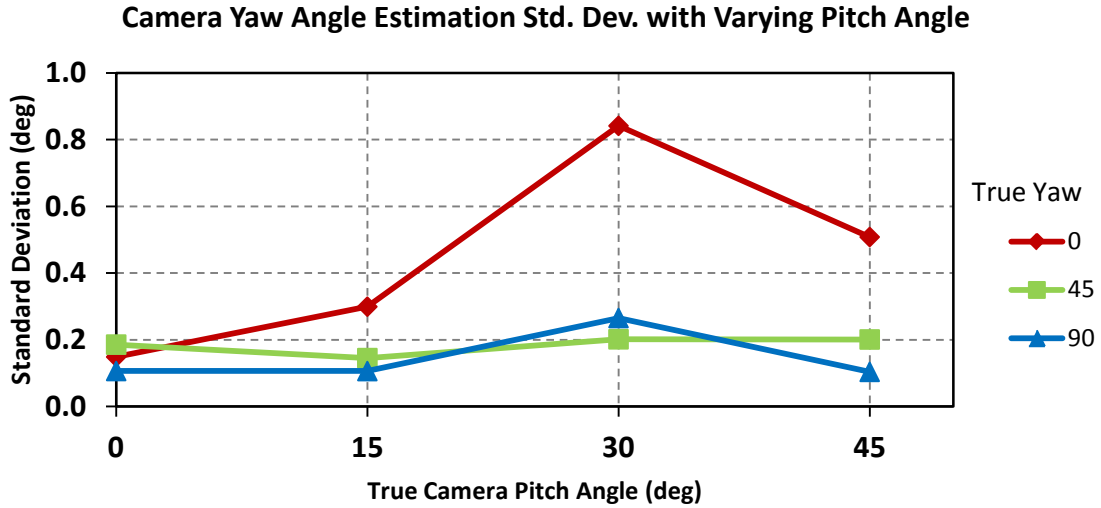
Figure 27. Camera yaw angle average estimation error across all trajectories.

With the camera orientation determined, the four trajectories are analyzed to determine which produced the most accurate estimation. Figure 28 through Figure 31 show estimation error and standard deviation broken down by trajectory and parameter for the best case camera orientation. The average errors across all of the maneuvers are low with very similar outliers. These results indicate that, outside of a completely straight and level trajectory, the exact path of the aircraft does not have an impact on the calibration estimation. Moreover, all of the camera orientations testing in this section showed the same trend; all four maneuvers producing very similar results.

Overall, a camera orientation 15 degrees pitch, and 90 degrees yaw was found to be the best case orientation, and the specific aircraft maneuver did not play a vital role in determining the accuracy of the estimated parameters.

60
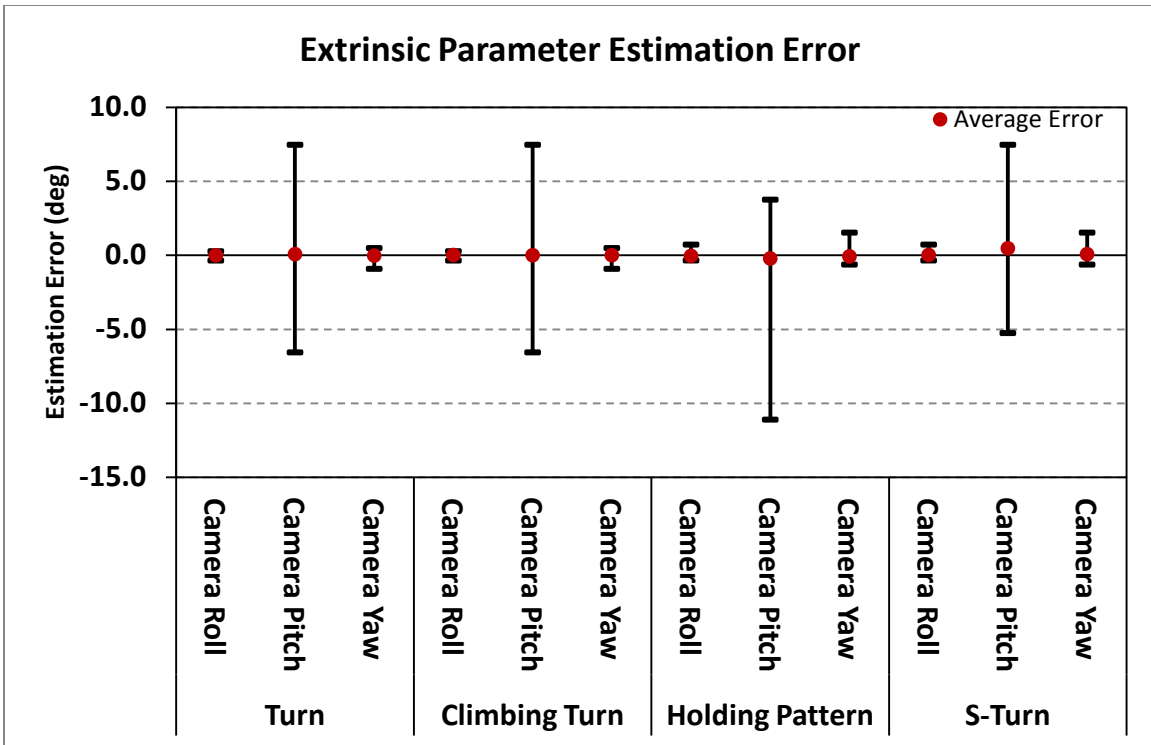
Figure 28. Extrinsic parameter estimation error for the best case camera orientation. Bars indicate maximum and minimum average error.
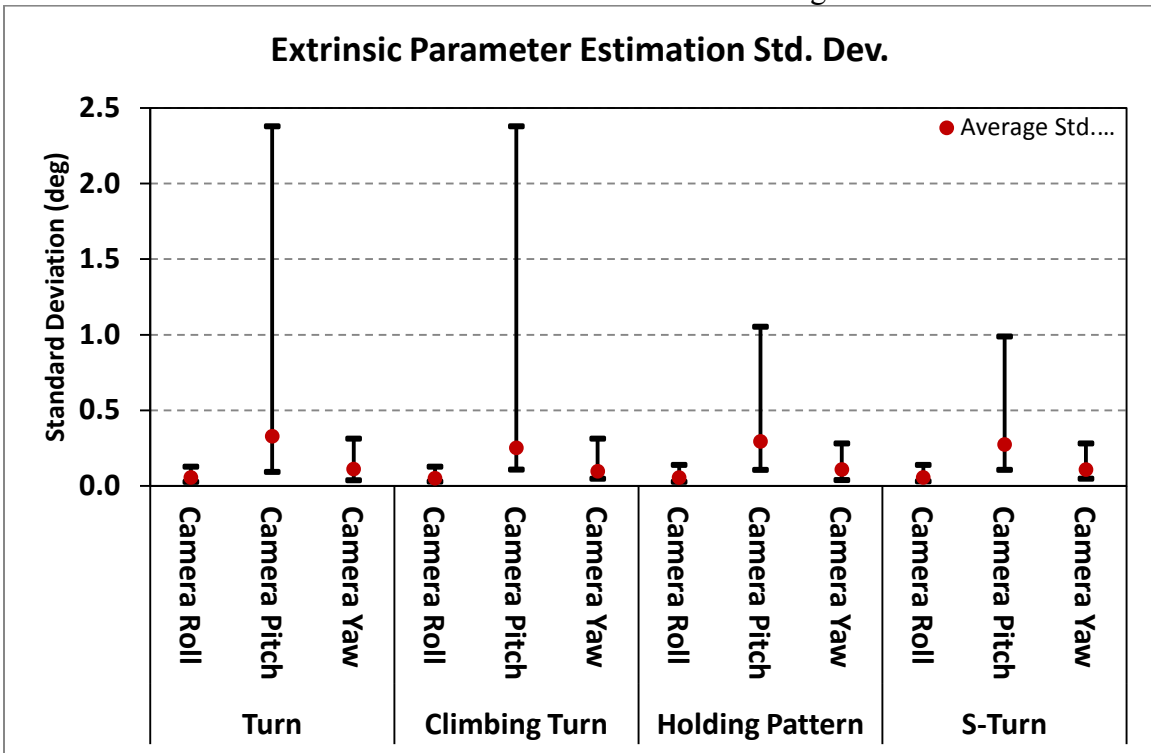
Figure 29. Extrinsic parameter estimation standard deviation for the best case camera orientation. Bars indicate maximum and minimum average standard deviation.
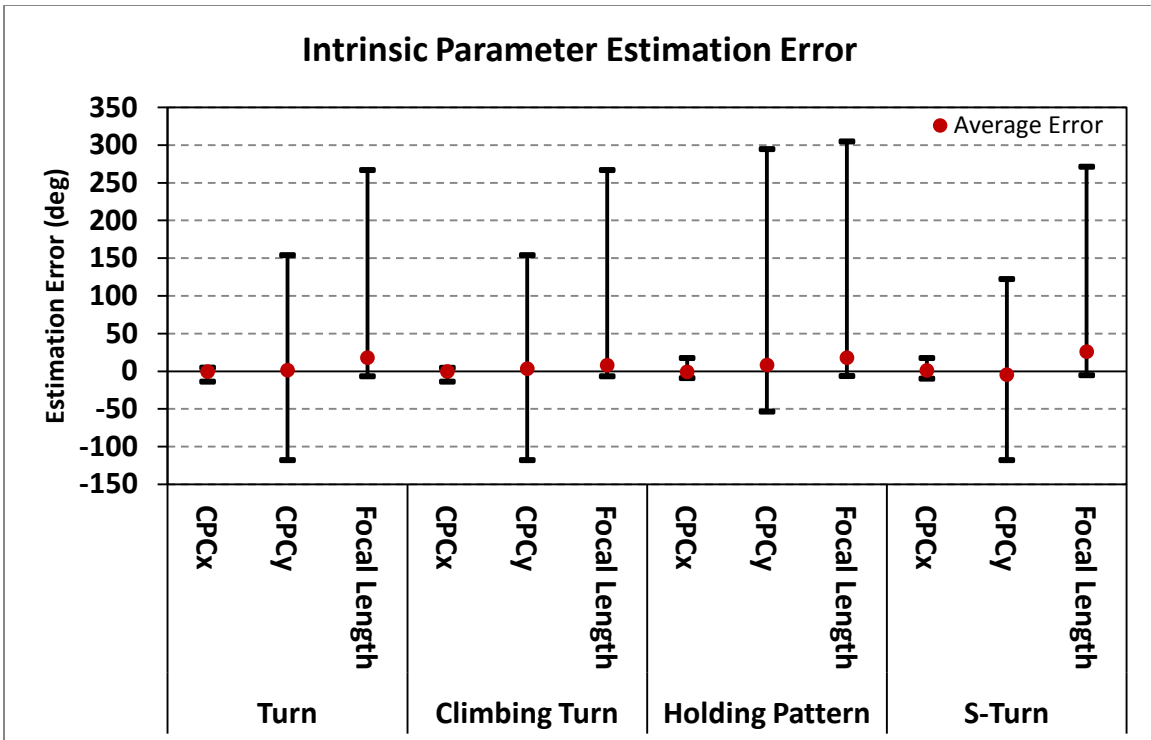
Figure 30. Intrinsic parameter estimation error for the best case camera orientation. Bars indicate maximum and minimum average error.
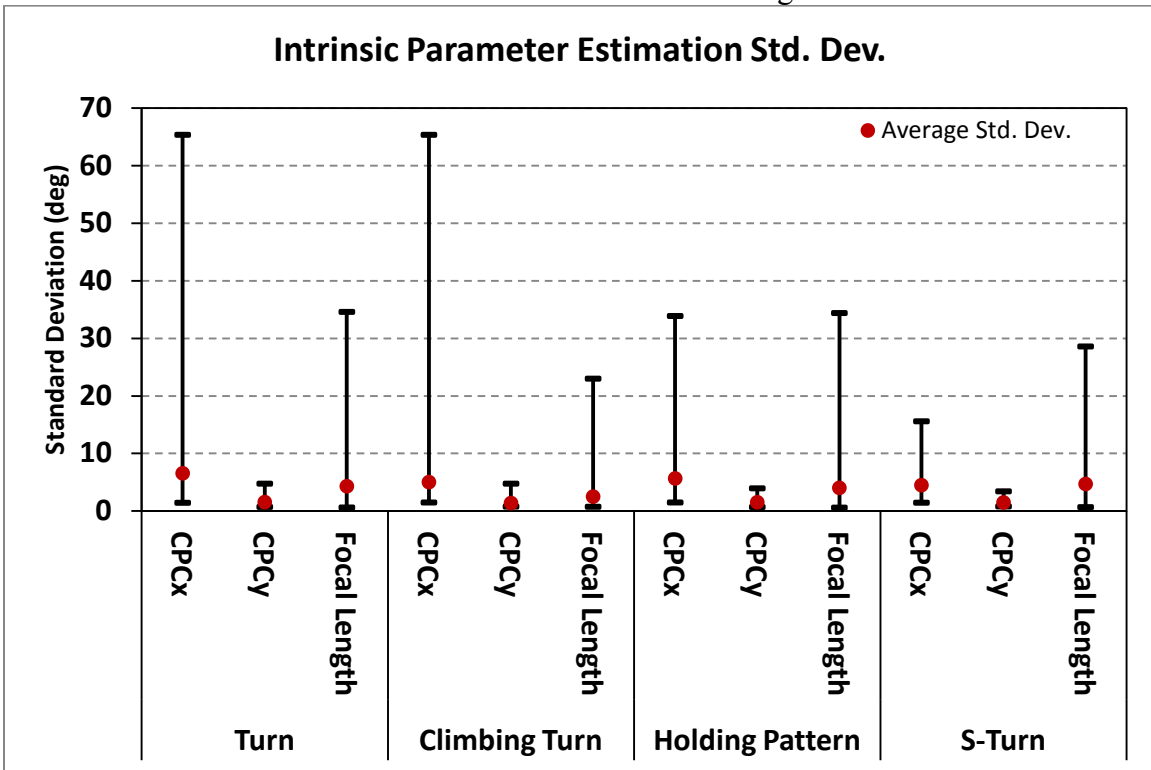
Figure 31. Intrinsic parameter estimation standard deviation for the best case camera orientation. Bars indicate maximum and minimum average standard deviation.

Flight Test

Flight test methodology and equipment are next presented along with data collection and post processing. The experimental flight test portion of this thesis was conducted as part of the Have SURF test management project (TMP) at the United States Air Force Test Pilot School (USAF TPS). Approximately 15.5 hours of flight testing were flown over seven sorties on a modified C-12C aircraft within the Precision Impact Range Area (PIRA) at Edwards AFB, CA, the R-2515 airspace at Edwards AFB, CA, the R-2517/34 airspace at Vandenberg AFB, CA and the airspace over the strip in Las Vegas, NV. All sorties were flown from 5-14 September 2012. The Responsible Test Organization for this project was the 412th Test Wing. The executing organization was the USAF TPS Have SURF test team. The test team consisted of three student test pilots, two student flight test engineers, and one student flight test combat systems operator. All experimental data presented in this report were gathered from this flight test.

The flying segment sought to collect data using maneuvers determined to be the "best case" of the simulation results. Two camera orientations were chosen and data was collected during a series of maneuvers. Multiple altitudes were used in order to vary the possible feature density of the images. Data collected was post processed using SURF and the automated calibration algorithm to provide a comparison with the simulation results.

Flight Test Equipment.

The flight test equipment used to collect image and trajectory data consisted of a digital camera mounted on the underside of a C-12C aircraft as well as a standalone

integrated GPS/INS.   Data monitoring and collection was controlled via a laptop computer. The position and attitude output from the GPS/INS was time correlated to the collected images in post processing to provided precise position and attitude information at each instance for the algorithm.

Digital Camera.

The imaging sensor was Prosilica GE-1660 2-megapixel digital camera fitted with a VS Technology Corporation SV-0614H 6.1 mm lens.   Images were transferred to an onboard laptop computer via an Ethernet network.   Images were collected as monochrome 8-bits per pixel at a resolution of 1600 by 1200 and saved in the Portable Gray Map (.pgm) format to conserve disk space. The camera also incorporated an input which allowed the SPAN to trigger the shutter to allow for correlation of the aircraft space and position data.

Integrated GPS/INS.

A Novatel SPAN-SE with an HG1700 AG58 Inertial Measurement Unit (IMU) was used to provide time, space, and position information (TSPI) of the aircraft. Further, the SPAN allowed for the correlation and logging of the TSPI data at the same instant in time in which an image was captured.   The SPAN featured a tight integration of a NovAtel GNSS receiver and the IMU. The SPAN provided continuous navigation information, using an Inertial Navigation System (INS), to bridge short Global Navigational Satellite Systems (GNSS) outages. Figure 32 and Figure 33 show the average navigation solution standard deviations as reported by the SPAN during data collection. The overall average horizontal and height standard deviations were 0.37 meters and 0.47 meters respectively.

Attitude error statistics were not contained in any of the data streams collected during testing; however Novatel reports accuracies of 0.018 degrees RMS in pitch and roll and 0.53 degrees RMS in azimuth. Two data streams were collected: a five Hertz direct sample of the attitude and navigation solution and a four Hertz sample stream that was collected concurrently with the images. The image correlated four Hertz data stream was used in data reduction with the automated calibration routine.
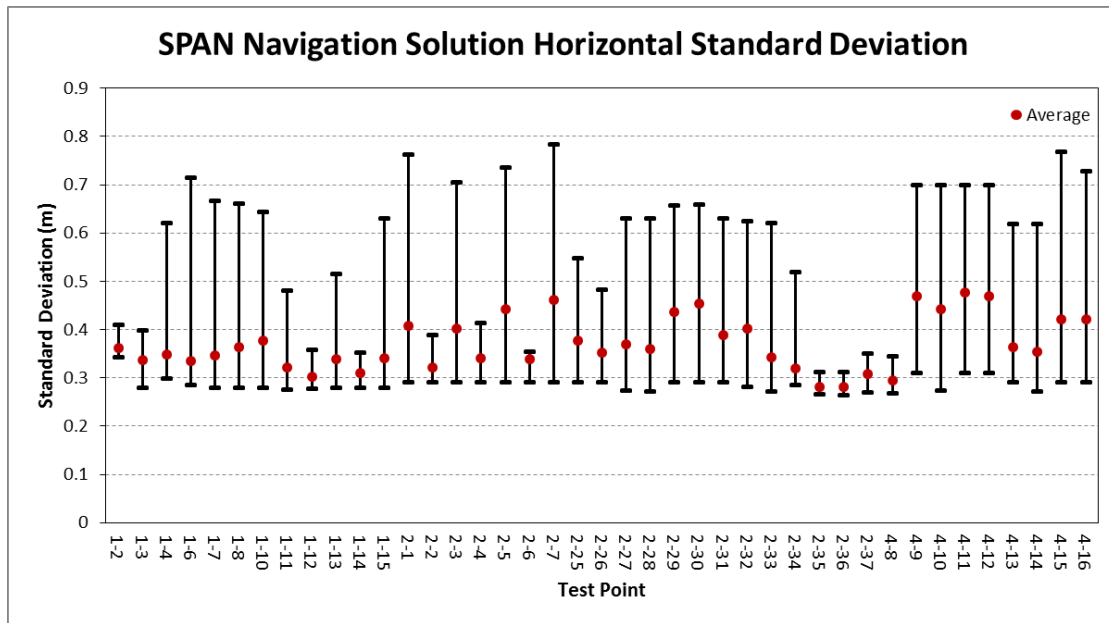
Figure 32. SPAN Navigation Solution Horizontal Standard Deviation. Error bars indicate the maximum and minimum values recorded during each test point.
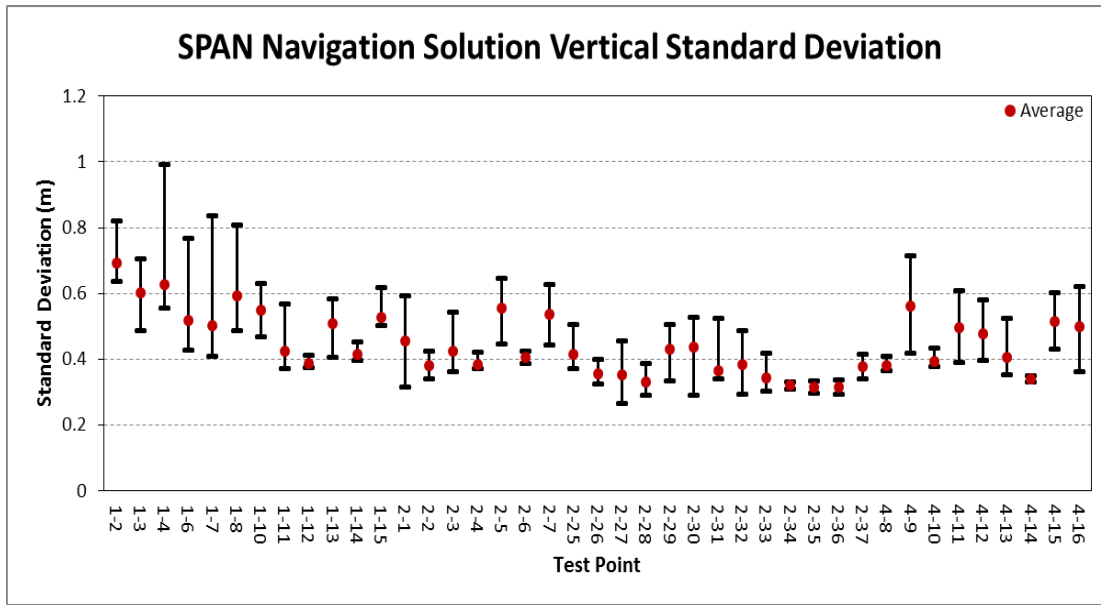
Figure 33. SPAN Navigation Solution Vertical Standard Deviation. Error bars indicate the maximum and minimum values recorded during each test point.

Test Aircraft.

Both the imaging sensor and the SPAN were mounted on board a C-12C Huron (Figure 34). The C-12C is a medium weight twin engine turbo-prop aircraft. The size of the aircraft allowed for the installation of the data collection equipment with relatively little modification to the airframe. The camera was mounted though the existing drift meter port on the right forward underside of the fuselage. A custom fabricated mounting bracket was used to hold the camera in place (Figure 35).

Figure 34. C-12C Huron

Figure 35. Prosilica camera Installed in the mounting bracket on the underside of the C-12C

The best and worst case camera orientations from the simulation results were chosen to gather comparison flight data. The worst case orientation was with the camera facing forward in the direction of the nose with an approximate 45 degree depression angle. The best case orientation was chosen to be facing to the right in the direction of the right wing with a pitch angle of 55 degrees. To avoid removing the cameras from the mounting assembly and preserve boresight measurements, two separate cameras were used. Each camera was contained in a separate mounting bracket with only one camera being mounted at a time. The SPAN and IMU as well as a power supply and a network switch were mounted on a single rigid tray in the cabin of the aircraft and are shown in Figure 36. The IMU was mounted commensurate with the conventions in the Novatel user manual and appropriate rotation and offset measurements were input in to the data collection software.
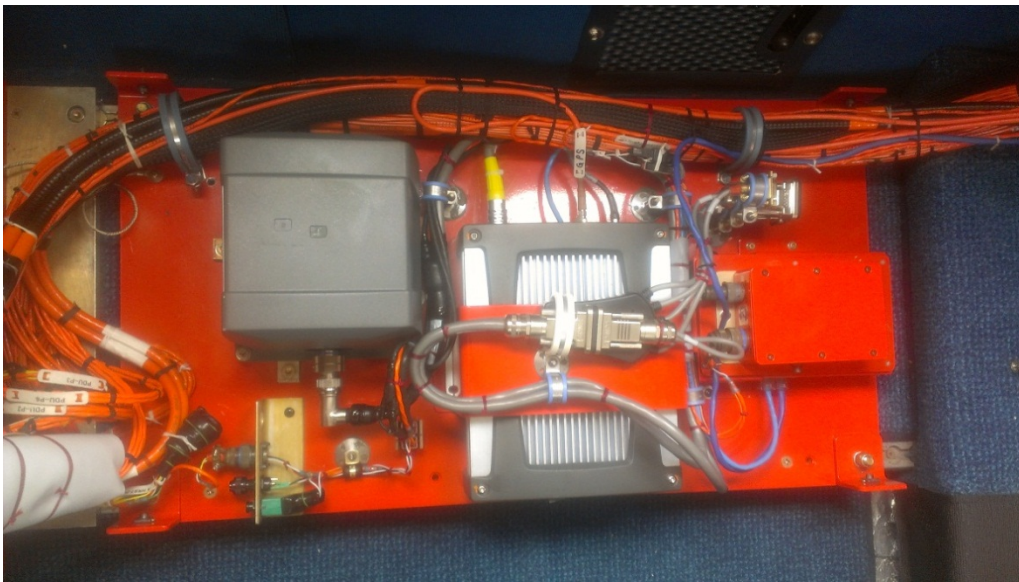
Figure 36. SPAN-SE receiver and HG1700 IMU mounted in the aircraft.

Baseline Calibration and Boresight.

A baseline calibration and boresight of the mounted camera was done to obtain data with which to compare the results of the flight test data. The intrinsic camera parameters and radial distortion coefficients were derived using the "Camera Calibration Toolbox for MATLAB©" [**15**]. This process is well known and will not be discussed in depth. Figure 37 shows a sample set of the grid data input into the calibration software. Several sets of calibration images were taken before each flight. Ground calibration results for each day are shown below.

Table 4. Sideward Camera Ground Calibration

Parameter	Side Camera 6 Sept 2012		Side Camera 11 Sept 2012		Sideways Camera 11 Sept 2012	
	Calibrated	Std. Dev.	Calibrated	Std. Dev.	Calibrated	Std. Dev.
g_x (pixels)	797.34	2.01	802.52	1.37	797.21	1.35
g_y (pixels)	629.15	4.83	605.09	2.22	604.97	2.36
g_f (pixels)	1123.88	3.92	1117.99	2.51	1128.34	2.71
$g_{\rho1}$ (none)	-0.003	0.003	-0.019	0.098	-0.004	0.094
$g_{\rho2}$ (none)	-0.004	0.0062	0.064	0.170	0.015	0.168
Parameter	Side Camera 11 Sept 2012		Side Camera 12 Sept 2012		Sideways Camera 13 Sept 2012	
	Calibrated	Std. Dev.	Calibrated	Std. Dev.	Calibrated	Std. Dev.
g_x (pixels)	789.37	1.33	805.22	1.43	805.78	4.74
g_y (pixels)	610.81	2.65	639.45	3.06	639.1	8.22
g_f (pixels)	1122.09	3.16	1080.55	3.20	1062.78	12.91
$g_{\rho1}$ (none)	-0.022	0.102	-0.010	0.098	-0.010	0.012
$g_{\rho2}$ (none)	0.085	0.202	0.019	0.150	0.020	0.029

Table 5. Forward Camera Ground Calibration Values

Parameter	Forward Camera 7 Sept 2012	
	Calibrated	Std. Dev.
g_x (pixels)	790.23	2.70
g_y (pixels)	634.37	4.28
g_f (pixels)	1131.93	6.06
$g_{\rho 1}$ (none)	-0.002	0.003
$g_{\rho 2}$ (none)	0.003	0.006
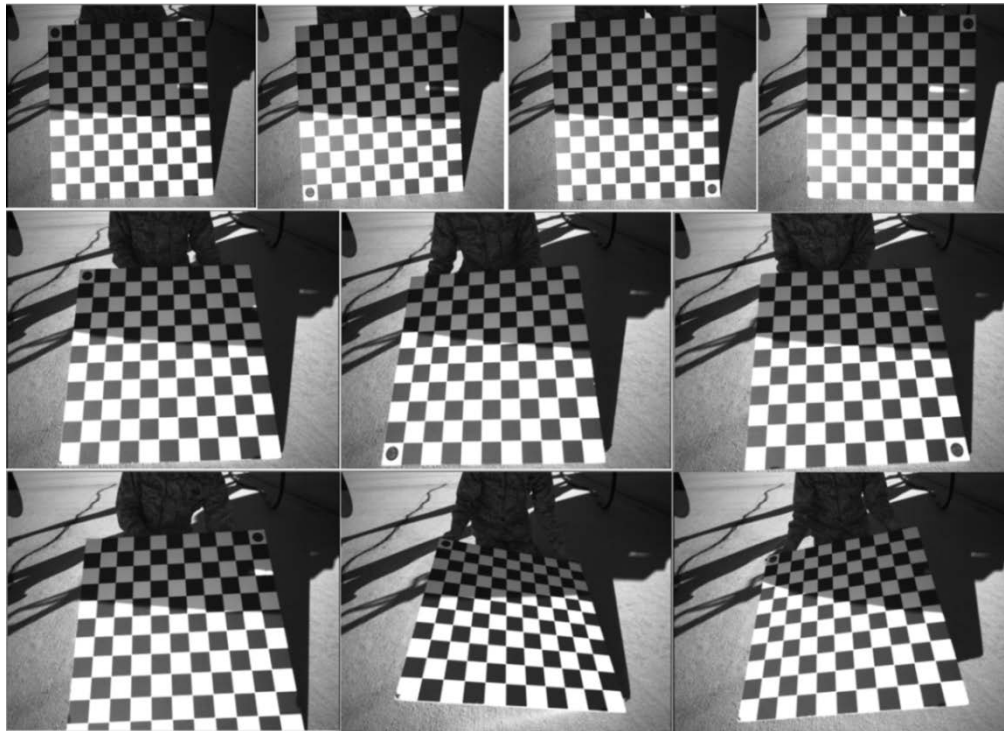
Figure 37. Calibration Board Images

A boresight was conducted using a Faro Ion high precision IFM-based laser tracker with an eight foot 'Platinum' arm to derive the extrinsic parameters of the camera as well as the IMU installation rotations. Camera boresight measurements were taken with respect to the camera case as installed in the bracket and a projection to the imaging sensor was done in the surveying software. The rotation and translation vectors obtained

were in reference to the aircraft body frame and were accurate to 0.1 degree and 0.1 mm. A single boresight was done prior to beginning flight testing and was considered to be accurate for the duration of the test. Boresight values are shown in Table 6.

Table 6. Boresight Values

Parameter	Forward Camera	Sideways Camera
g_ϕ (deg)	-4.0902	-4.2604
g_θ (deg)	45.7463	53.5822
g_ψ (deg)	3.9854	94.2604

Flight Test Methodology.

From simulation, the automated calibration algorithm showed that changes to camera orientation and aircraft trajectory as well as measurement noise affected the accuracy of the calibration estimates. During the flying portion of the testing, data were taken during four standard maneuvers over the same landscape in the vicinity of Cords Road and California City within the R-2515 airspace near Edwards AFB, CA. Both sideways and forward camera orientations were tested in order to gain the largest variety of landscape trajectory within the image plane.

The C-12 was flown through preplanned patterns at three baseline altitudes for each maneuver type: 10,000 feet, 5,000 feet, and 2,000 feet AGL. The nominal airspeed for all maneuvers was 150 ±10 KIAS. Photos and corresponding TSPI data were taken at a 4Hz rate. The executed maneuvers at each altitude were:

- Constant bank 360° degree turn – A 30° bank turn into the direction of the camera through a full 360°. Image collection was started once established in the turn and stopped while still in the banked turn.

71

- Constant bank 360° climbing turn – Executed the same as above except with a 500 fpm climb rate.
- Holding pattern – A standard holding pattern with 30 second straight legs and 180° turns using 30° angle of bank at each end. Image collection was started with straight segments and completed after the second 180° turn.
- S-turn – Two 30° bank turns with a reversal after 90° of heading change. The turn transitions were made with smooth aileron inputs. Image collection was started right before establishing the initial turn and ceased once the initial heading was reached following the turn reversal.

For efficiency, test points using a straight and level trajectory were not flown. Instead the data from the straight portions of the holding patterns was used. Overall the executed test points flown are listed in Table 7.

Table 7. Test Points Flown

Camera Orientation	Altitude (ft AGL)	Maneuver	Number of Test Points
Sideways	10000	360° Turn	3
Sideways	5000	360° Turn	2
Sideways	2000	360° Turn	1
Sideways	10000	360° Climbing Turn	3
Sideways	2000	360° Climbing Turn	1
Sideways	10000	Holding Pattern	3
Sideways	5000	Holding Pattern	1
Sideways	2000	Holding Pattern	1
Sideways	10000	S-Turns	4
Sideways	5000	S-Turns	3
Sideways	2000	S-Turns	1
Forward	10000	360° Turn	2
Forward	5000	360° Turn	3
Forward	10000	360° Climbing Turn	2
Forward	5000	360° Climbing Turn	3
Forward	10000	Holding Pattern	2
Forward	5000	Holding Pattern	3
Forward	10000	S-Turns	2
Forward	5000	S-Turns	3

The same stipulations on the camera offset and distortion parameters were again placed on the flight test data during processing. In each case the threshold of the SURF algorithm was manually adjusted to allow for a sufficient number of features to be detected without overloading. A very low threshold was required to detect features when imaging the relatively featureless desert, while a larger threshold was required for populated areas. For maneuvers that included images of both desert and populated areas a compromise was required. This phenomenon is shown in Figure 38; two images from a Turn test point illustrate the effect of the threshold. In this case the threshold is set to allow for a moderate amount of features to be detected in the desert; however the amount detected in the populated area is extreme.
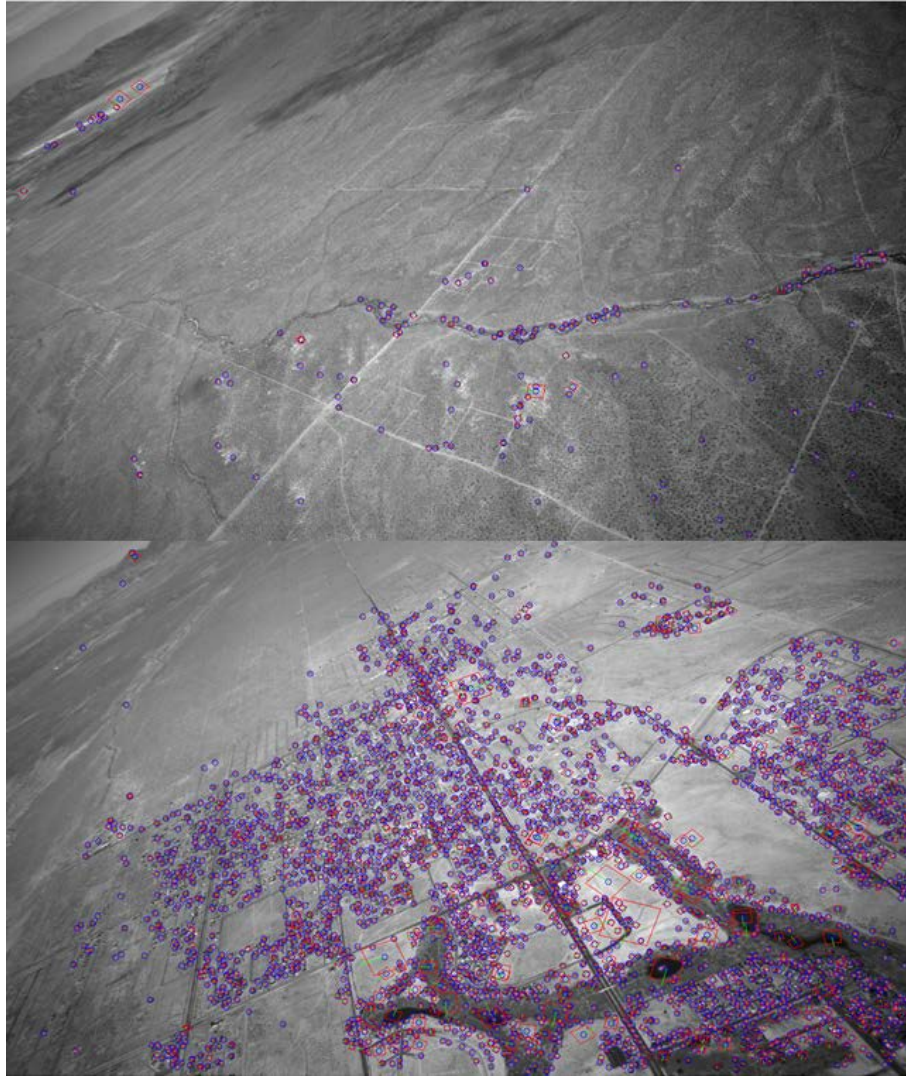
Figure 38. SURF features detected using a fixed threshold showing the need for threshold tuning.

The best results from the calibration routine were obtained when the algorithm was provided data at the full four hertz collection rate. Data processed at frequencies lower than four hertz tended to produce fewer feature correlations across image sets.

Flight Test Results.

In this section the results from each maneuver are discussed. Due to the variability of the distortion parameter results from the ground calibration process (Table 4), this

analysis focuses on the accuracy of the camera rotation angles and CPC. Each case is discussed along with interpretation of nonstandard outcomes.

Turn Results.

A total of eleven turn test points were flown; five with the camera in the sideward orientation and six in the forward orientation. Parameter estimation errors for the turn maneuvers are shown in Figure 39 and Figure 40. The turning maneuvers produced consistent plausible results for both camera orientations. Each of the sideward camera test points resulted in a pitch calibration within one degree and yaw within three degrees. The forward camera results were slightly worse in that the pitch results were all within two degrees and yaw within four degrees. Of note, neither camera orientation was able to accurately reproduce the roll angle. Errors of up to five degrees were seen for this and all of the remaining maneuvers. This is an interesting result, as the simulation was able to match the roll angle calibration exactly in all of the cases except the straight and level configuration.
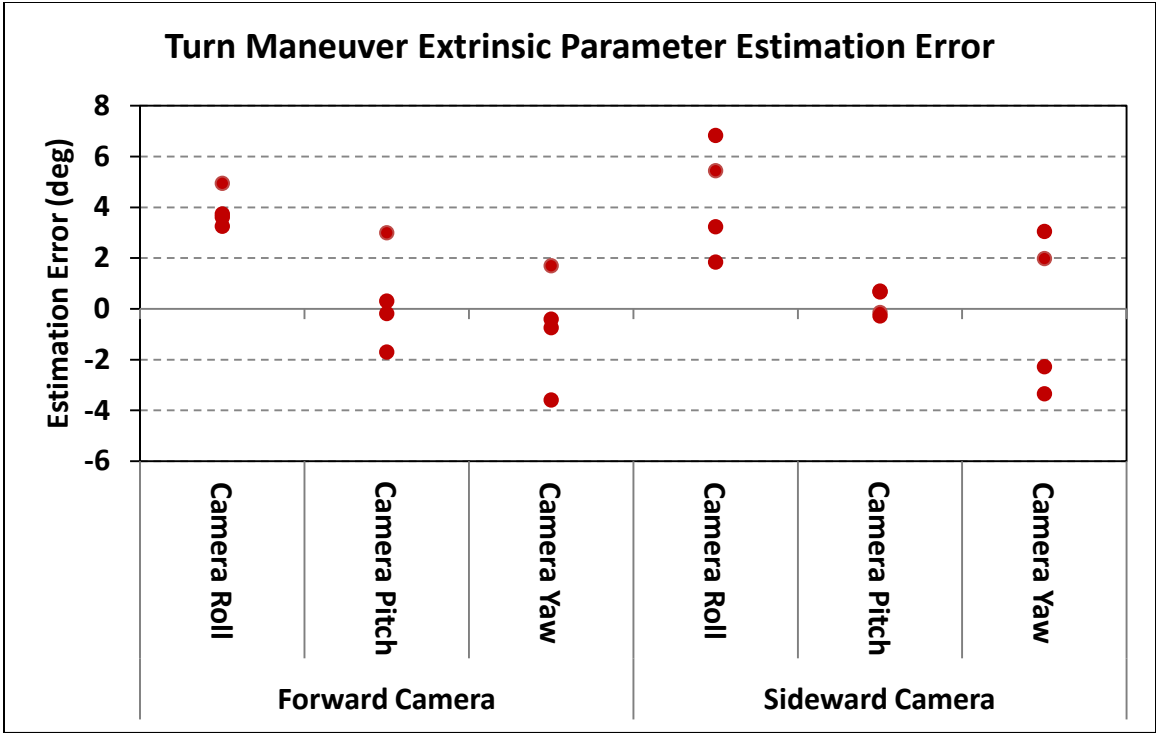
Figure 39. Turn Maneuver Extrinsic Parameter Estimation Error
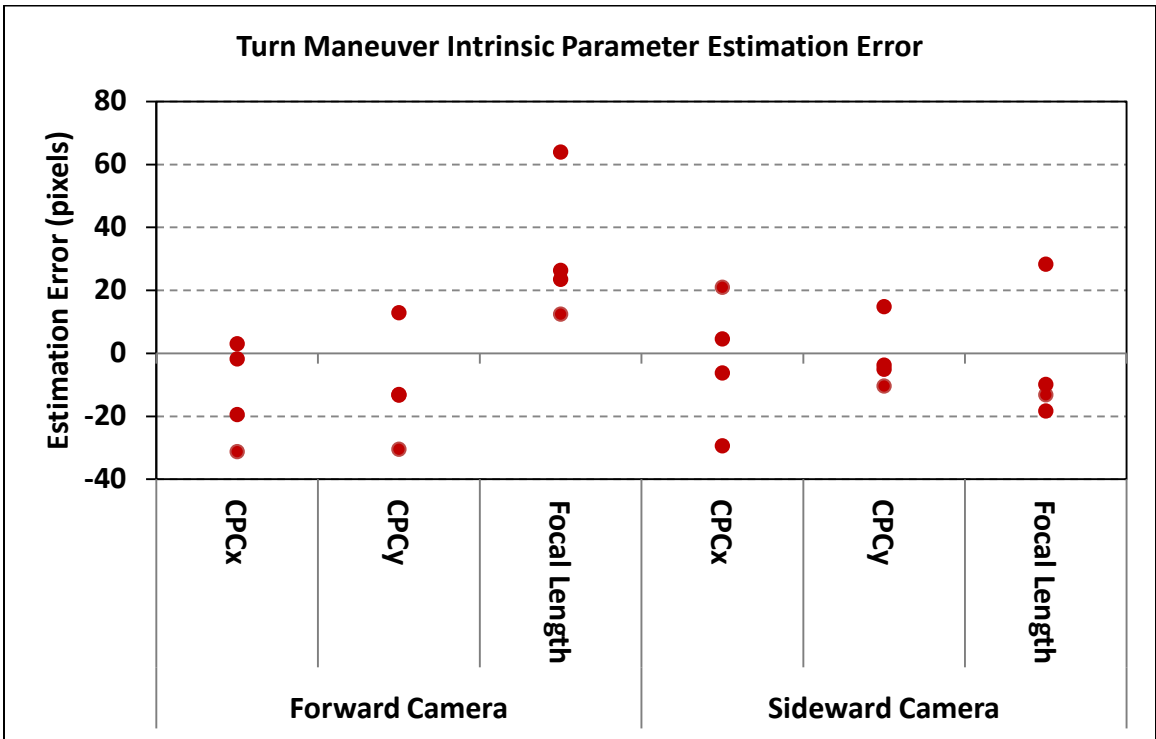
Figure 40. Turn Maneuver Intrinsic Parameter Estimation Error

Climbing Turn Results.

Nine climbing turn test points were flown; four sideward and five forward. Parameter estimation errors for the climbing turn maneuvers are shown Figure 41 and Figure 42. The climbing turn maneuvers produced extrinsic parameter estimates very similar to the turning results; however, the intrinsic parameters were noticeably off. Compared to the level turn maneuver points, the addition of the aircraft pitch during the maneuver did not have apparent effect on the extrinsic parameter estimates for either camera orientation - they in fact follow the same trend. The intrinsic parameters show a tendency for CPC y coordinate and focal length to have greater estimation errors with the climbing turn than with the level turn.
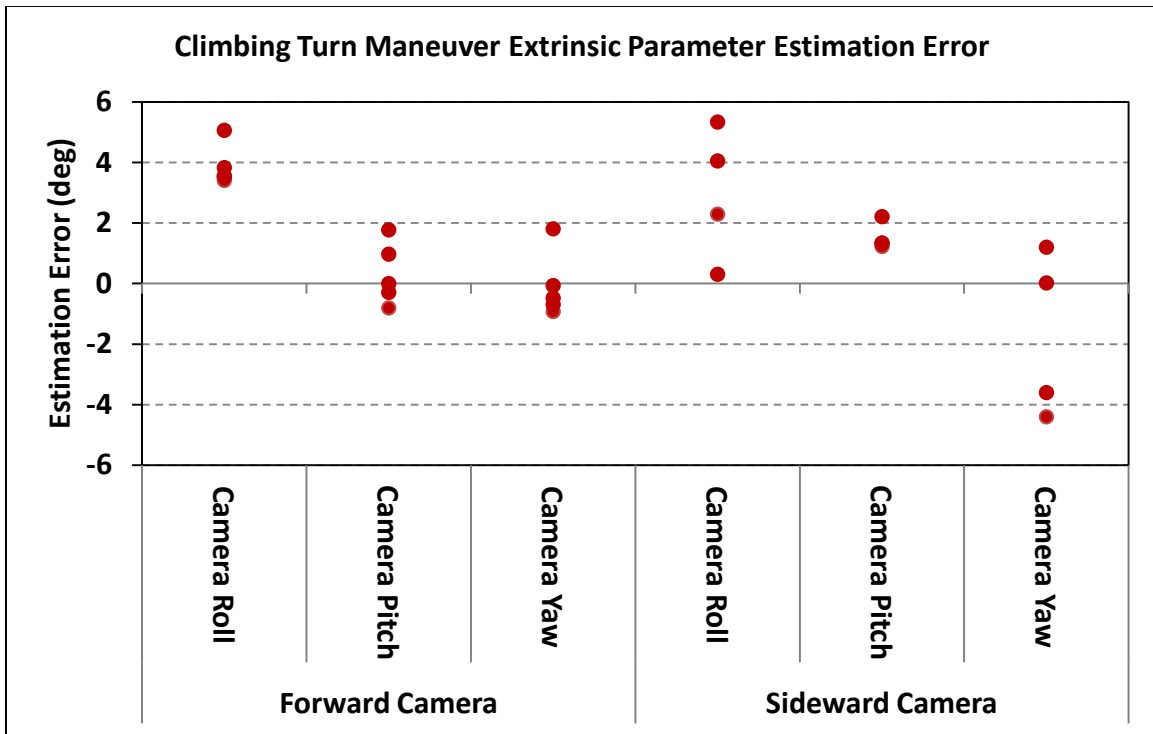
Figure 41. Climbing Turn Maneuver Extrinsic Parameter Estimation Error
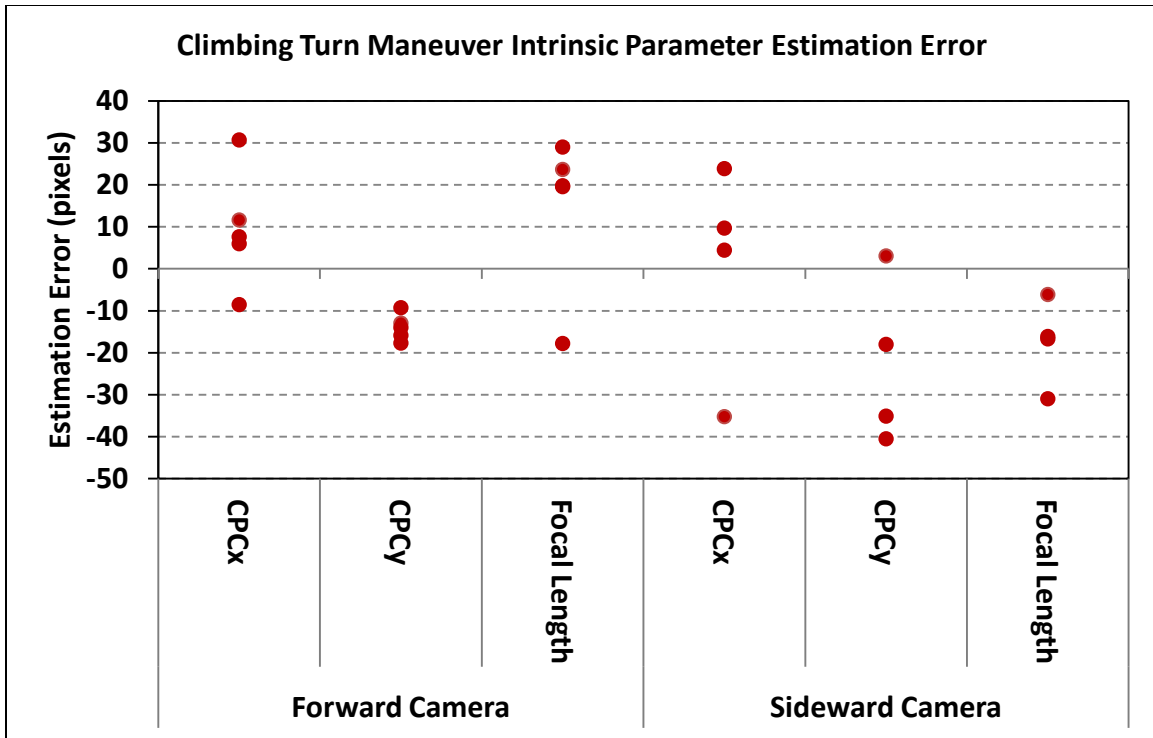
Figure 42. Climbing Turn Maneuver Intrinsic Parameter Estimation Error.

Holding Pattern Results.

Ten holding pattern test points were flown, five of each orientation. Average parameter estimation errors for the hold pattern maneuvers are shown Figure 43 and Figure 44. Once again the results follow the same trend as in the turning test point maneuvers. The magnitude of both extrinsic and intrinsic parameter estimation errors are roughly the same as that in the turning maneuver. Camera roll angle is still shown to have the greatest error for both camera orientations. Overall, the forward camera orientation generated better parameter estimates then the sideward camera.
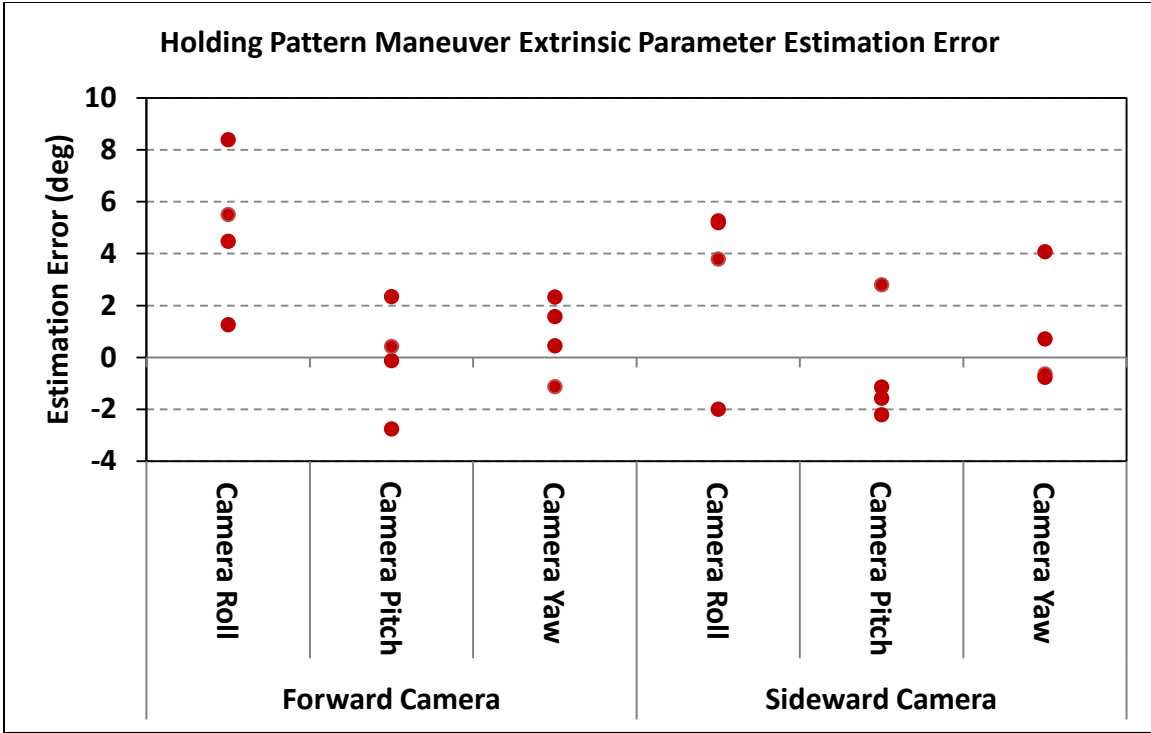
Figure 43. Holding Pattern Maneuver Extrinsic Parameter Estimation Error
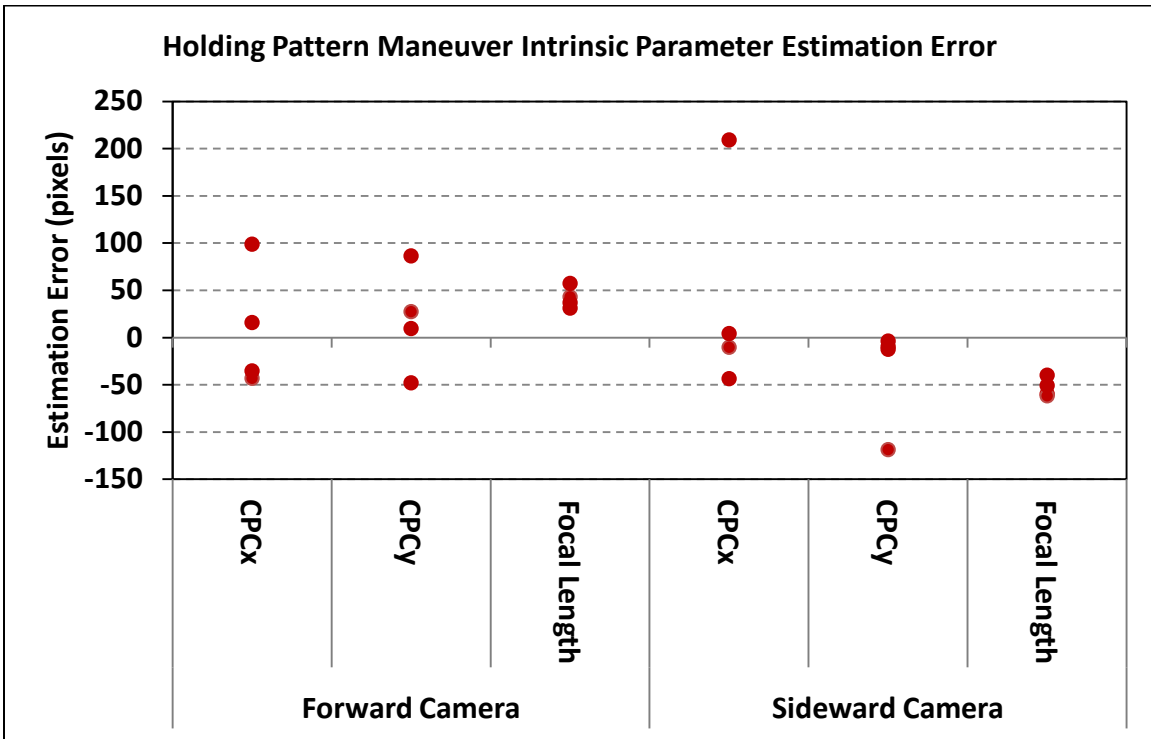
Figure 44. Holding Pattern Maneuver Intrinsic Parameter Estimation Error

Straight Results.

Much like the simulation results, the straight maneuvers were the most detrimental to the calibration routine. Data from the straight portion of the holding pattern test points were extracted and used for these test points. A total of ten test points were run; five with the sideward orientation and five forward. Parameter estimation errors for the straight maneuvers are shown in Figure 45 and Figure 46. The results for the sideward camera orientation tended to have more accurate results than the forward orientation. This follows with the simulation straight maneuver results in that as the yaw angle approaches 90 degrees estimated yaw and roll error decrease while the pitch error is roughly constant (Figure 14 through Figure 15).
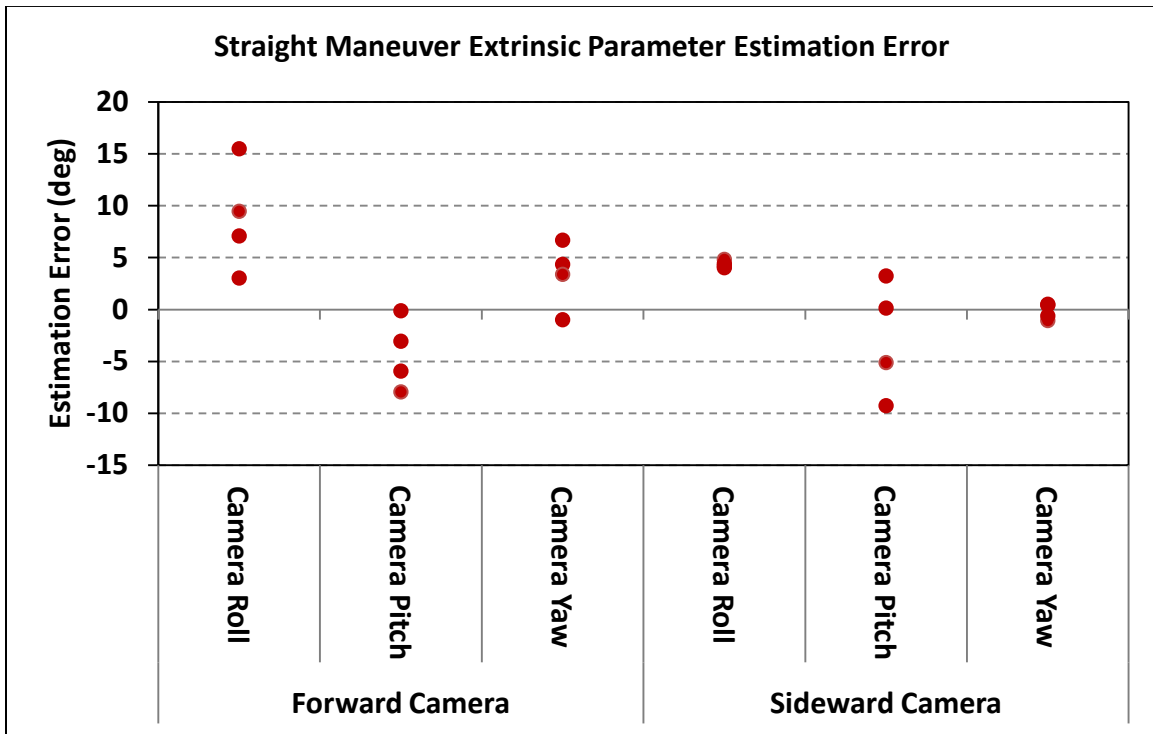
Figure 45. Straight Maneuver Extrinsic Parameter Estimation Error
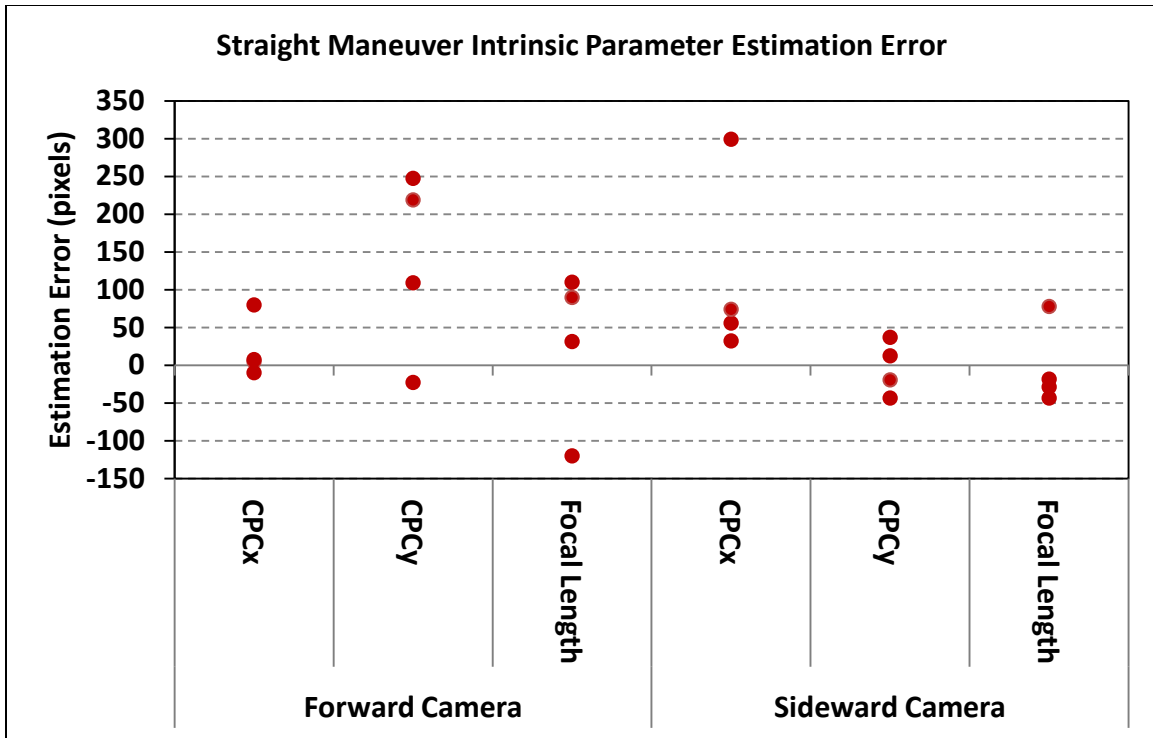
Figure 46. Straight Maneuver Intrinsic Parameter Estimation Error

S-Turn Results.

Eleven S-Turn test points were run: six with the sideward orientation and five forward. Average S-turn parameter estimation results are shown in Figure 47 and Figure 48. The sideward camera orientation produced a wide range of calibration parameters for both extrinsic and intrinsic parameters. Of the five forward camera results, two were outside a plausible range, though each of those was more reasonable than any of the sideward camera results. The variances in the sideward camera results may be attributed to the look angle of the camera. As the aircraft was in the right bank, the camera was looking down at the ground, however as the aircraft went into a left turn the camera was imaging the horizon. Projections done during the left turn have a very high graze angle

from the camera to the ground on the horizon. This would lead to even small angles inducing large errors into the solution.
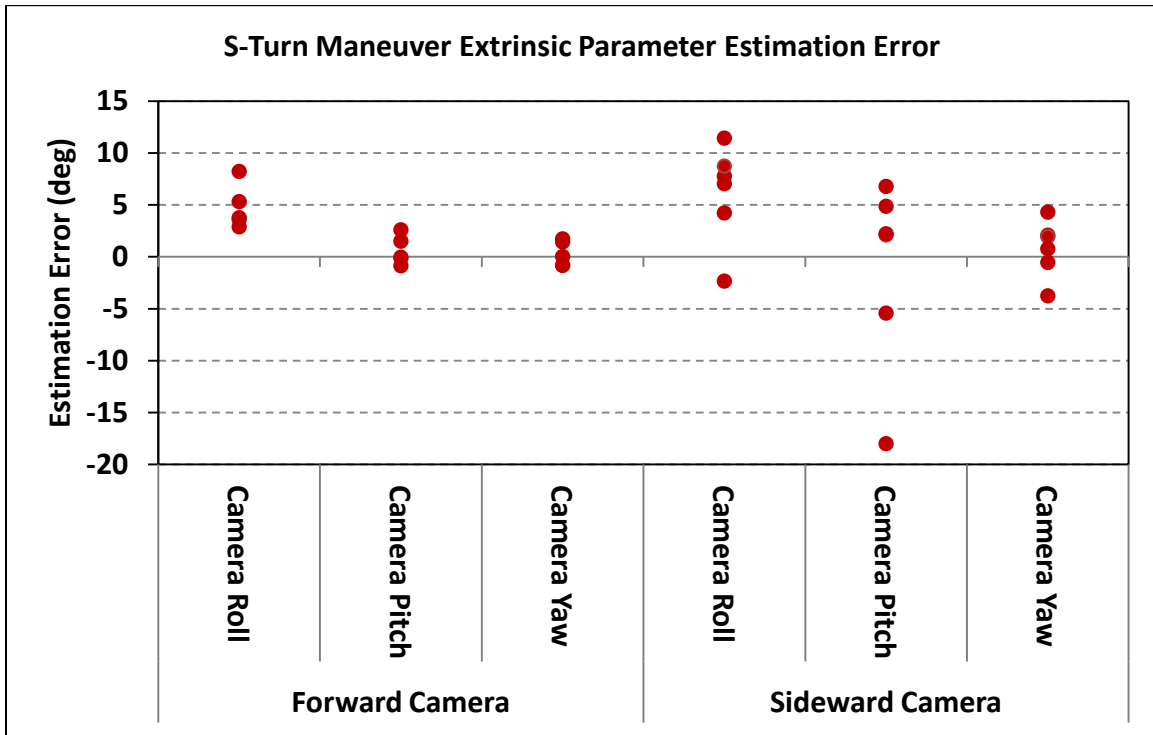
Figure 47. S-Turn Maneuver Extrinsic Parameter Estimation Error
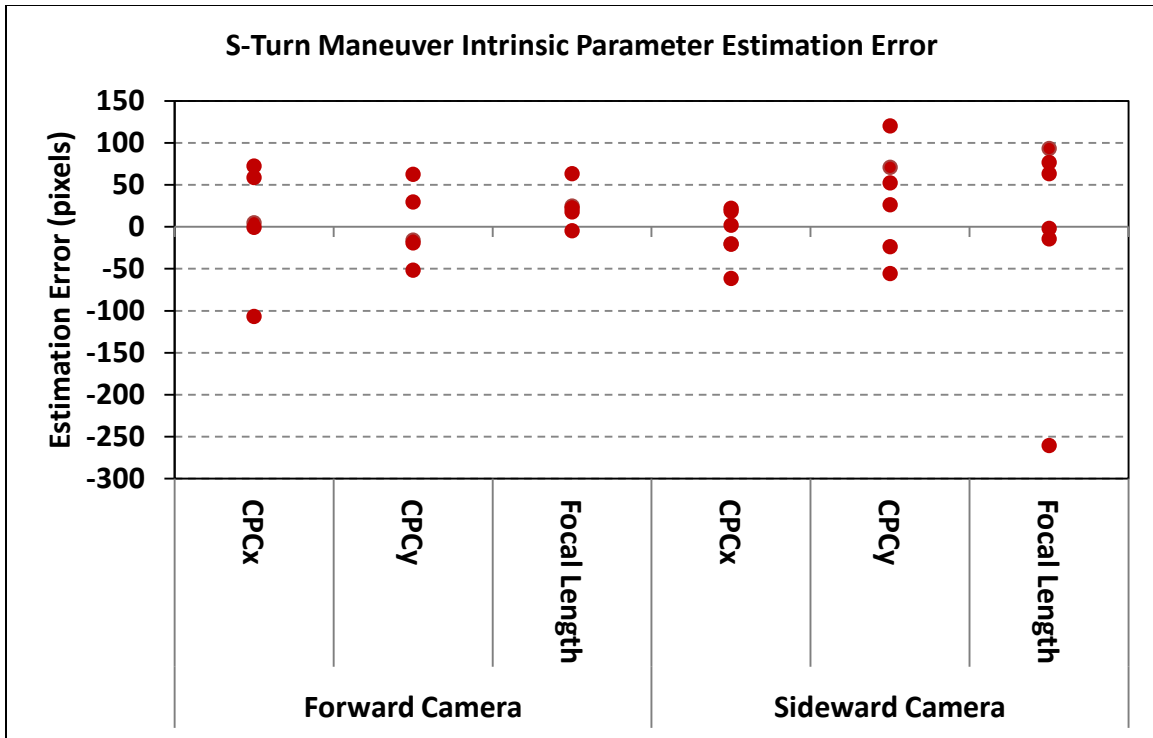
Figure 48. S-Turn Maneuver Intrinsic Parameter Estimation Error

Simulation with Flight Test Data

Given the results of both the simulation and the flight test a final comparison combining the flight test trajectory data and the simulation generated feature data was conducted. The TSPI data from each test point was input into the simulation as the aircraft trajectory. The feature generation and correlation portion of the simulation software was then run given the trajectory using the ground calibration and boresight values as truth data. These results allow for the removal of any inaccuracies or inconsistencies due to indefinite feature pixel location and improper feature matching.

Maneuver Results.

Much like the results from the simulation test cases, the combined simulation and flight test cases produced almost universally accurate calibrations due to the perfect

measurements. All but one of the trajectories, for both camera orientations produced accurate results. Selected results for the straight trajectories are presented in Table 8. Test point 2-3st was the only case that did not estimate the exact correct results. Of note, in all other cases the aircraft bank angle shifted from zero to 2.5 degrees of bank over the course of the segments. In test point 2-3st the maximum bank angles reached during the 22.75 second run was 1.03 degree which lasted for only one second, with an average of 0.33 degrees. These results corroborate the results of the bank angle simulation test cases in that even very small bank angle changes tended to provide the algorithm with enough information to converge to an accurate calibration result.

Table 8. Selected Combined Simulation and Flight Test Straight Maneuver Results

Parameter	Forward Camera Test Point 2-7st				Forward Camera Test Point 2-3st			
	True	Initial	Error	Std. Dev.	True	Initial	Error	Std. Dev.
g_ϕ (deg)	-4.09	0	0	0.428	-4.09	0	1.09	0.188
g_θ (deg)	45.75	42	0	0.292	45.75	42	5.05	0.195
g_ψ (deg)	-2.99	0	0	0.320	-2.99	0	1.7	0.129
g_x (pixels)	790.23	800	0	6.664	790.23	800	20.3	3.967
g_y (pixels)	634.37	600	0	4.352	634.37	600	93.2	2.767
g_f (pixels)	-1131.93	-1000	0	6.480	-1131.93	-1000	93.0	4.059
$g_{\rho 1}$ (none)	-0.00234	0	0	0.002	-0.00234	0	0.003	0.002
$g_{\rho 2}$ (none)	0.00304	0	0	0.003	0.00304	0	0.010	0.001
Parameter	Side Camera Test Point 1-4st				Side Camera Test Point 1-7st			
	True	Initial	Error	Std. Dev.	True	Initial	Error	Std. Dev.
g_ϕ (deg)	-4.26	-1	0	0.042	-4.26	-1	0	0.067
g_θ (deg)	53.58	55	0	0.113	53.58	55	0	0.160
g_ψ (deg)	94.26	95	0	0.039	94.26	95	0	0.045
g_x (pixels)	805.22	800	0	1.943	805.22	800	0	2.022
g_y (pixels)	639.45	600	0	1.285	639.45	600	0	2.037
g_f (pixels)	-1080.45	-1000	0	2.315	-1080.45	-1000	0	2.155
$g_{\rho 1}$ (none)	-0.01007	0	0	0.002	-0.01007	0	0	0.002
$g_{\rho 2}$ (none)	0.01934	0	0	0.002	0.01934	0	0	0.002

Error Determination.

In an attempt to qualify a possible source of error in the flight test results several more test cases were run. The number and layout of detected features was tuned in the simulation to match that of the TSPI data from test point 1-3. Test point 1-3 was a turning maneuver where the entire series of images tracked essentially the same points on the ground. California City is visible in in roughly the same location in all of the images as seen in Figure 49.

Figure 49. Image from Test Point 1-3. The road intersection denoted with the dashed circle stays in approximately the same area in all images collected during this test point.

In the flight test data an average of 29 features per image were detected by SURF; the simulation was tuned to reproduce the same amount of features per simulated image. Then, the flight test estimated calibration parameters in conjunction with the simulation features were used to calculate a projected horizontal RMS error of 399.5 meters. This would simulate a zero noise solution with perfect feature matching. Five baseline cases were run; all of which resulted in zero parameter estimation and HRMS error. Navigation and attitude measurement noise was then added with standard deviations of 0.33 meters and 0.2 degrees in pitch and roll and 0.5 degrees in azimuth to match that of the SPAN data for this test point. Feature noise was then gradually added in an attempt to match the flight test estimated parameters and projected RMS error. All cases were

run with the same initial conditions as the flight test data. Five runs of each case in Table 9 were done; results are shown in Figure 50.

Table 9. Flight/Simulation Test Cases.

Case	Position Meas. Noise Std. Dev. (meters)	Pitch and Roll Meas. Noise Std. Dev. (deg)	Azimuth Meas. Noise Std. Dev. (deg)	Pixel Meas. Noise Std. Dev. (pixels)
1	0	0	0	0
2	0.33	0.02	0.05	0
3	0.33	0.02	0.05	1
4	0.33	0.02	0.05	2
5	0.33	0.02	0.05	3
6	0.33	0.02	0.05	4
7	0.33	0.02	0.05	5
8	0.33	0.02	0.05	6
9	0.33	0.02	0.05	7
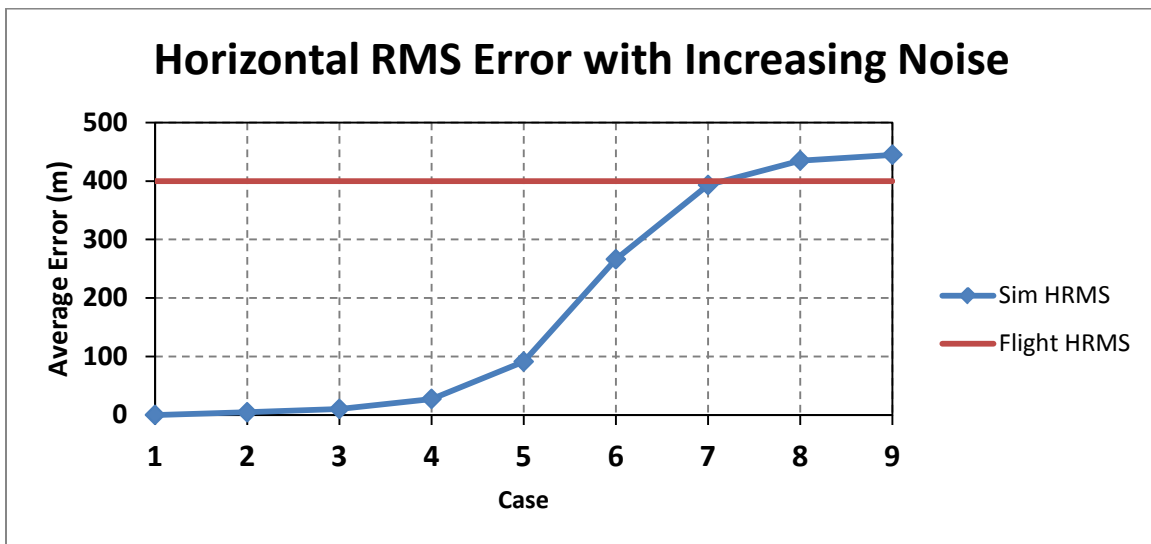
Figure 50. Test Point 1-3 Simulation Average Horizontal Projection Error with Increasing Noise and Few Features

At five pixels of image feature pixel noise, the average HRMS had reached the magnitude of the flight test data. However, the features matched across each image at five pixels had reduced below the threshold of the flight test data. Figure 51 and Figure 52 show the 60 features in view across the most images in the flight test data and simulation data with noise respectively. It can be seen that the features in the simulation

data with noise are not correlated across as many features as in the flight test data. Given

this result, a second approach was taken.
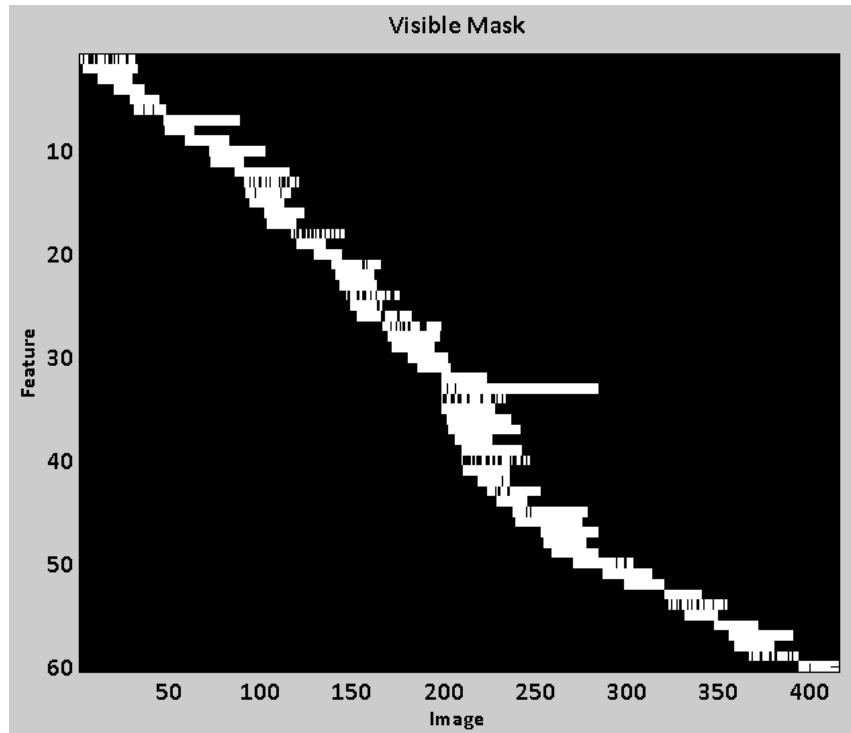
Figure 51. Visible Features from Flight Test Point 1-3
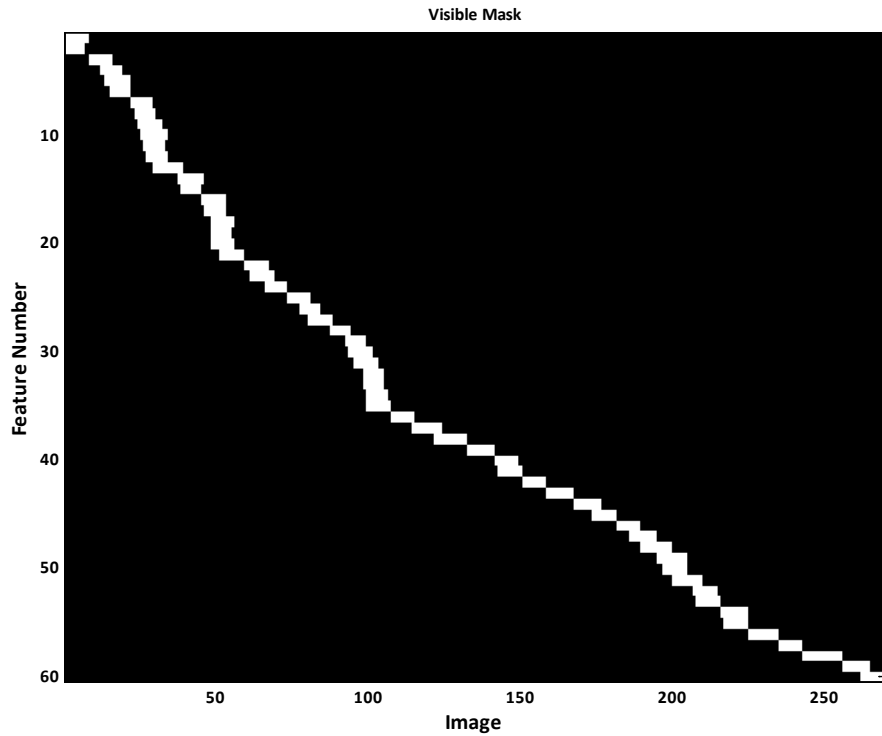
Figure 52. Visible Features from Simulation of Test Point 1-3 with 5 pixel measurement noise

The second approach was taken wherein the features per image were set to the maximum number of 79 from the flight test data. The same trend in the feature visibility was seen in this data. Projection error results were on par with that found previously, and reached the threshold of the actual flight test data at five pixels of noise.
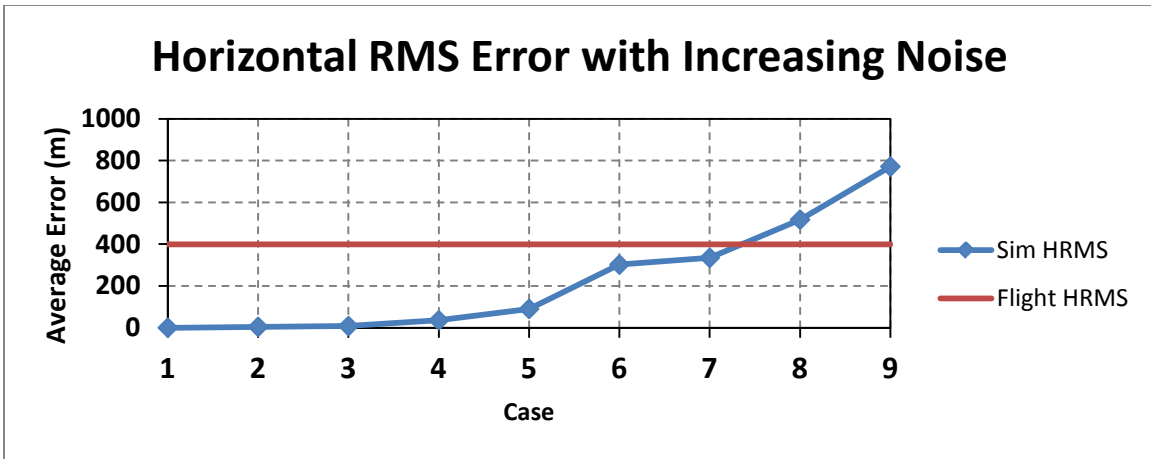
Figure 53. Test Point 1-3 Simulation Average Horizontal RMS Error with Increasing
Noise and Many Features

Given these results, the conclusion can be drawn that feature correlation is a factor in
the accuracy of the calibration parameter estimates. Even with turning SURF to prevent
egregious overloading of features, several of the images had upwards of one thousand
features detected in them. Given such a high feature density it is plausible that several
mismatched of features could occur. Features detected very close together could be
miscorrelated by the algorithm as the same feature in successive images. Overall, these
results do show that the noise in the navigation solution would not lead to the errors seen
in the flight test results and that a combination of noise in the feature detection algorithm
and/or feature mismatches are the most likely source of the errors.

Summary

A case study for three different data scenarios was presented along with methodology
for the flight test conducted. The simulation results showed that the largest factors
affecting the quality of accurate calibration estimation is the bank angle of the aircraft
and noise in the feature detection process. The maneuver results show that if the aircraft

had less than two degrees of bank, the calibration estimates are sub-standard. Further, inaccurate results were seen by injecting noise in to the simulated image pixel locations of detected features. A turn maneuver with a camera orientation of zero roll, 45 degrees pitch and ninety degrees yaw was determine to be the best case using the simulation with realistic measurement noise. Flight test data showed results that matched those of the simulation. Lastly, the simulation combined with the flight test aircraft TSPI data further validated the result that the straight and level maneuver and image noise could produce results that corresponded to those seen in the flight test data.

V. Conclusions and Recommendations.

This thesis developed and analyzed a method for automatically estimating the parameters needed to calibrate an aircraft mounted camera. An image to world frame vector projection was developed using the constraints inherent to an image sensor mounting on an aircraft. Several key features to the baseline algorithm presented in [1] were added including radial distortion and a key reference frame rotation. The projection methodology was used in a bundle adjustment routine to estimate the key calibration parameters of a camera given a combination of precise aircraft TSPI data and features detected in time correlated imagery. A simulator was developed and implemented to test the calibration routine's sensitivity to various conditions. Sensitivity to aircraft maneuver, measurement noise, and initial conditions was conducted. Finally, a flight test was conducted and results compared to the simulation analysis. This chapter summarizes the results of the simulation and flight test and draws conclusions on each. Recommendations for future work are made based on the results.

Conclusions

The results exhibited in this research clearly show the influence that aircraft maneuver and noise have on the calibration parameter estimation accuracy of a bundle adjustment type calibration routine in the aerial environment. The "perfect" simulation results display a requirement of the routine to have visibility in at least the bank and azimuth parameters of the aircraft, which are coupled. Further, it was shown that during the simulated straight and level maneuver, the parameter estimate error increased as the camera orientation pitch angle increased but decreased as the roll and yaw angles were

increased. Simulation of both navigation system measurement noise and feature detection noise was done. The error in the calibration estimates due to navigation system noise was found to be relatively minor compared that of the feature noise. Given the accuracy of modern integrated GPS/INS navigation systems, the projection error induced by the navigation noise was shown to be in the tenths of meters. Feature detection noise proved to be much more detrimental to accurate parameter estimation. Projection errors in the hundreds of meters were observed with noise amounts as small as one to three pixels. Errors due to initial conditions did not manifest until highly unrealistic values were used. Overall, the largest impact to accurate parameter estimates was the feature measurement noise.

The flight test data agreed with the simulation results. Though the results from each maneuver did not produce an exact calibration, the trend of each maneuver matched that of the simulation. An exact match to the ground calibration results was not expected, as the inflight dynamics of the aircraft would tend to alter the parameters. The straight maneuver was shown to have greater errors in the calibration parameters than the other maneuvers. A further validation of the algorithm was done using the flight test collected aircraft position and attitude data with the features generated in the simulator. Much like the pure simulation results, these results reiterated that aircraft bank angle is a contributing factor to accurate parameter estimates and that the amount of the bank required is contingent on the look down orientation of the camera. Lastly, an attempt was made to replicate the errors seen in the flight test results. It was shown that errors approaching that of the flight test results could be replicated by adding navigation and feature noise.

Recommendations for Future Work

This research focused on several basic conditions in which to perform a sensitivity analysis. Several areas for additional development and study are available to be accomplished. The separate areas of additional work can be addressed: the calibration algorithm, further simulation cases and flight test collected data.

Though the automated calibration algorithm was shown to be successful in calculating a highly accurate set of camera calibration parameters no effort was made to increase the efficiency of the algorithm or introduce known surveyed features. The sparse matrices generated during the bundle adjustment process offer many areas for increase calculation speed. Moreover, the largest bottleneck of the process occurred during the feature detection process with SURF. The implementation of a more efficient feature detection algorithm would vastly improve the overall calculation speed. The addition of the option for the algorithm to accept a set of known surveyed features along with the detected features could vastly improve the confidence of the results. The bundle adjustment algorithm has the ability to set an observation as an exact match. Currently this option would require manual manipulation of the data.

One important case that was not covered in this work is that of the possible interdependence of the calibration parameters. An evaluation could be made by setting the intrinsic or extrinsic parameters to known values and iterating on the other.

The flight test portion of the research offers much in the way of future work. The largest area for future work would be the introduction of another feature detection routine. Though SURF provided satisfactory features, an interesting case study could be made comparing different feature detection software. To further characterize the

calibration results, a projection using surveyed points and the calibrated values could be done to numerically characterize the error. Overall, the data collected during the flight test is considered to be accurate and could be used in further investigation of this or other algorithms.

Appendix A - Data

Realistic Simulator Results

Flight Test Results

Table 10. Sideward Camera, Turn Maneuver Test Point Results

Test Point		1-3	4-8	4-12	4-15
Altitude (ft AGL)		10000	2000	5000	5000
Airspeed (KIAS)		150	150	150	150
surf Threshold		0.006	0.006	0.006	0.0006
surf Octaves		5	5	5	5
surf Step		3	3	3	3
Number of Images		434	287	218	323
Number of Features		60	60	60	60
LM Iteration Time		173.2	148.2	184.2	180.5
LM Iterations		31	8	15	13
g_ϕ (Roll)	Calibrated	-2.42	1.18	-1.03	2.57
	Std. Dev.	0.294	0.020	0.059	0.039
	Initial	-1	-1	-1	-1
g_θ (Pitch)	Calibrated	54.28	53.44	53.30	54.25
	Std. Dev.	0.159	0.044	0.098	0.057
	Initial	55	55	55	55
g_ψ (Yaw)	Calibrated	90.92	96.24	91.98	97.31
	Std. Dev.	0.232	0.018	0.060	0.040
	Initial	95	95	95	95
g_x	Calibrated	791.10	826.76	776.39	810.35
	Std. Dev.	3.173	0.748	1.928	1.024
	Initial	800	800	800	800
g_y	Calibrated	624.08	628.70	653.92	635.34
	Std. Dev.	4.274	0.378	0.935	0.655
	Initial	600	600	600	600
g_f	Calibrated	-1094.18	-1075.95	-1072.66	-1081.10
	Std. Dev.	4.135	0.442	0.951	0.640
	Initial	-1100	-1100	-1100	-1100
$g_{\rho 1}$	Calibrated	-0.033	0.010	-0.063	0.022

	Std. Dev.	0.0225	0.0014	0.0042	0.0036
	Initial	0	0	0	0
$g_{\rho2}$	Calibrated	0.028	-0.016	0.073	-0.067
	Std. Dev.	0.0296	0.0022	0.0069	0.0087
	Initial	0	0	0	0

Table 11. Sideward Camera, Climbing Turn Maneuver Results

Test Point		1-11	1-12	1-13	4-11
Altitude (ft AGL)		10000	10000	10000	2000
Airspeed (KIAS)		150	150	150	150
SURF Threshold		0.0002	0.0006	0.0006	0.006
SURF Octaves		5	5	5	5
SURF Step		3	3	3	3
Number of Images		261	261	442	384
Number of Features		60	60	60	60
LM Iteration Time		176.9	143.9	239.2	176.7
LM Iterations		23	15	25	13
g_ϕ (Roll)	Calibrated	-0.21	-1.97	-3.96	1.07
	Std. Dev.	0.188	0.113	0.144	0.021
	Initial	-1	-1	-1	-1
g_θ (Pitch)	Calibrated	55.79	54.81	54.91	54.92
	Std. Dev.	0.211	0.088	0.157	0.052
	Initial	55	55	55	55
g_ψ (Yaw)	Calibrated	94.28	89.86	90.66	95.46
	Std. Dev.	0.176	0.073	0.119	0.026
	Initial	95	95	95	95
g_x	Calibrated	807.05	762.13	821.22	810.23
	Std. Dev.	2.689	1.117	2.889	0.959
	Initial	800	800	800	800
g_y	Calibrated	594.06	632.22	611.15	598.60
	Std. Dev.	2.663	1.925	3.092	0.406
	Initial	600	600	600	600
g_f	Calibrated	-1153.45	-1128.60	-1138.61	-1079.46
	Std. Dev.	2.330	0.834	2.956	0.523
	Initial	-1100	-1100	-1100	-1100
$g_{\rho1}$	Calibrated	-0.069	0.099	-0.061	0.011
	Std. Dev.	0.0116	0.0067	0.0291	0.0017
	Initial	0	0	0	0
$g_{\rho2}$	Calibrated	0.137	-0.131	0.091	0.023
	Std. Dev.	0.0227	0.0123	0.0991	0.0030

	Initial	0	0	0	0

Table 12. Sideward Camera, Holding Patter Maneuver Results

Test Point		1-4	1-7	1-10	4-9	4-13
Altitude (ft AGL)		10000	10000	10000	2000	5000
Airspeed (KIAS)		150	150	150	150	150
SURF Threshold		0.0002	0.0006	0.0002	0.002	0.002
SURF Octaves		5	5	5	5	5
SURF Step		3	3	3	3	3
Number of Images		743	725	460	687	708
Number of Features		60	60	60	60	60
LM Iteration Time		487.8	579.0	301.6	162.4	342.9
LM Iterations		29	73	24	10	18
g_ϕ (Roll)	Calibrated	4.13	-6.26	-0.47	0.93	1.00
	Std. Dev.	0.033	0.139	0.081	0.010	0.022
	Initial	-1	-1	-1	-1	-1
g_θ (Pitch)	Calibrated	55.93	52.01	56.38	51.37	52.44
	Std. Dev.	0.032	0.115	0.125	0.041	0.026
	Initial	55	55	55	55	55
g_ψ (Yaw)	Calibrated	96.59	98.33	93.62	94.97	93.49
	Std. Dev.	0.014	0.035	0.036	0.008	0.009
	Initial	95	95	95	95	95
g_x	Calibrated	762.02	1006.58	795.50	810.01	762.25
	Std. Dev.	0.467	1.917	1.925	0.717	0.484
	Initial	800	800	800	800	800
g_y	Calibrated	581.20	619.68	520.43	626.67	635.40
	Std. Dev.	0.506	2.951	1.638	0.291	0.340
	Initial	600	600	600	600	600
g_f	Calibrated	-1085.50	-1182.21	-1124.49	-1113.55	-1102.62
	Std. Dev.	0.295	0.825	1.105	0.358	0.255
	Initial	-1100	-1100	-1100	-1100	-1100
$g_{\rho 1}$	Calibrated	0.189	-0.094	-0.196	-0.003	-0.104
	Std. Dev.	0.0022	0.0028	0.0077	0.0013	0.0015
	Initial	0	0	0	0	0
$g_{\rho 2}$	Calibrated	-0.208	-0.013	0.345	-0.040	0.153
	Std. Dev.	0.0036	0.0022	0.0174	0.0022	0.0028
	Initial	0	0	0	0	0

Table 13. Sideward Camera Straight Maneuver Results

Test Point		1-4_st	1-7_st	4-13_st	4-9_st
Altitude (ft AGL)		10000	10000	5000	2000
Airspeed (KIAS)		150	150	150	150
SURF Threshold		0.0002	0.0006	0.002	0.002
SURF Octaves		5	5	5	5
SURF Step		3	3	3	3
Number of Images		121	101	91	101
Number of Features		60	60	60	60
LM Iteration Time		374.9	278.6	196.8	183.4
LM Iterations		37	39	19	20
g_ϕ (Roll)	Calibrated	-0.24	0.56	0.15	-0.07
	Std. Dev.	0.129	0.212	0.083	0.031
	Initial	-1	-1	-1	-1
g_θ (Pitch)	Calibrated	53.72	48.47	56.81	44.31
	Std. Dev.	0.320	0.396	0.176	0.138
	Initial	55	55	55	55
g_ψ (Yaw)	Calibrated	94.76	93.19	93.64	94.68
	Std. Dev.	0.056	0.070	0.030	0.007
	Initial	90	90	95	90
g_x	Calibrated	853.32	871.54	838.14	1105.08
	Std. Dev.	5.973	4.506	2.273	1.308
	Initial	800	800	800	800
g_y	Calibrated	666.26	609.92	595.83	651.69
	Std. Dev.	3.363	5.447	2.506	1.952
	Initial	600	600	600	600
g_f	Calibrated	-1140.70	-1044.54	-1106.04	-1091.35
	Std. Dev.	4.244	2.967	1.383	0.778
	Initial	-1100	-1100	-1100	-1100
$g_{\rho 1}$	Calibrated	0.014	0.143	-0.041	0.049
	Std. Dev.	0.0034	0.0122	0.0020	0.0013
	Initial	0	0	0	0
$g_{\rho 2}$	Calibrated	-0.026	-0.137	0.018	-0.039
	Std. Dev.	0.0038	0.0123	0.0027	0.0011
	Initial	0	0	0	0

Table 14. Sideward Camera S-Turn Maneuver Results

Test Point		1-6	1-8	1-14	1-15	4-10	4-14
Altitude (ft AGL)		10000	10000	5000	5000	2000	5000
Airspeed (KIAS)		150	150	150	150	150	150
SURF Threshold		0.0006	0.0006	0.0002	0.0006	0.0006	0.0009
SURF Octaves		5	5	5	5	5	5
SURF Step		3	3	3	3	3	3
Number of Images		315	296	259	251	226	204
Number of Features		60	60	60	60	60	60
LM Iteration Time		249.6	327.5	324.0	163.0	762.7	217.9
LM Iterations		36	53	46	26	82	15
g_ϕ (Roll)	Calibrated	3.51	4.48	2.78	-6.60	-0.03	7.17
	Std. Dev.	0.061	0.042	0.045	0.071	0.015	0.042
	Initial	-1	-1	-1	-1	-1	-1
g_θ (Pitch)	Calibrated	60.36	55.69	55.80	48.16	35.58	58.45
	Std. Dev.	0.063	0.057	0.058	0.067	0.075	0.045
	Initial	55	55	55	55	55	55
g_ψ (Yaw)	Calibrated	96.34	96.20	95.03	90.50	93.72	98.57
	Std. Dev.	0.032	0.024	0.019	0.021	0.014	0.038
	Initial	95	95	95	95	95	95
g_x	Calibrated	799.30	777.21	776.67	819.46	824.56	744.35
	Std. Dev.	0.736	0.527	0.860	0.909	0.925	0.903
	Initial	800	800	800	800	800	800
g_y	Calibrated	573.52	700.10	655.46	749.39	691.47	615.47
	Std. Dev.	0.892	0.646	0.691	1.135	0.589	0.673
	Initial	600	600	600	600	600	600
g_f	Calibrated	-1059.17	-1029.15	-1124.32	-1045.74	-1077.10	-1323.26
	Std. Dev.	0.488	0.432	0.486	0.508	0.480	0.680
	Initial	-1100	-1100	-1100	-1100	-1100	-1100
$g_{\rho 1}$	Calibrated	0.201	0.251	0.128	-0.121	-0.117	-0.264
	Std. Dev.	0.0032	0.0025	0.0033	0.0020	0.0011	0.0039
	Initial	0	0	0	0	0	0
$g_{\rho 2}$	Calibrated	-0.222	-0.261	-0.207	0.101	0.020	0.649
	Std. Dev.	0.0046	0.0032	0.0070	0.0023	0.0015	0.0097
	Initial	0	0	0	0	0	0

Table 15. Forward Camera, Turn Maneuver Results

Test Point		2-28	2-32	2-37	2-33
Altitude (ft AGL)		5000	10000	10000	10000
Airspeed (KIAS)		150	150	150	150
SURF Threshold		0.0006	0.0004	0.0004	0.0002
SURF Octaves		5	5	5	5
SURF Step		3	3	3	3
Number of Images		379	443	427	422
Number of Features		60	60	60	60
LM Iteration Time		315.6	261.8	495.2	441.2
LM Iterations		43	25	60	44
g_ϕ (Roll)	Calibrated	-0.48	0.85	-0.84	-0.35
	Std. Dev.	0.103	0.054	0.052	0.077
	Initial	0	0	0	0
g_θ (Pitch)	Calibrated	45.56	48.74	44.05	46.05
	Std. Dev.	0.213	0.109	0.145	0.205
	Initial	45	45	45	45
g_ψ (Yaw)	Calibrated	3.24	5.68	0.40	3.58
	Std. Dev.	0.199	0.125	0.140	0.194
	Initial	0	0	0	0
g_x	Calibrated	793.26	758.99	770.77	788.45
	Std. Dev.	3.692	1.546	1.797	2.092
	Initial	800	800	800	800
g_y	Calibrated	621.08	603.89	647.26	621.28
	Std. Dev.	3.050	2.169	2.037	2.457
	Initial	600	600	600	600
g_f	Calibrated	-1107.86	-1118.91	-1067.43	-1105.01
	Std. Dev.	3.653	1.648	1.934	2.367
	Initial	-1100	-1100	-1100	-1100
$g_{\rho 1}$	Calibrated	0.009	-0.010	-0.015	0.002
	Std. Dev.	0.0032	0.0025	0.0018	0.0022
	Initial	0	0	0	0
$g_{\rho 2}$	Calibrated	-0.001	0.005	0.004	0.009
	Std. Dev.	0.0051	0.0030	0.0015	0.0027
	Initial	0	0	0	0

Table 16. Forward Camera, Climbing Turn Maneuver Results

Test Point		2-2	2-6	2-29	2-34	2-35
Altitude (ft AGL)		5000	5000	5000	10000	10000
Airspeed (KIAS)		150	150	150	150	150
SURF Threshold		0.0006	0.0006	0.0006	0.0001	0.0002
SURF Octaves		5	5	5	5	5
SURF Step		3	3	3	3	3
Number of Images		397	414	409	477	511
Number of Features		60	60	60	60	60
LM Iteration Time		181.5	263.6	250.6	262.5	375.5
LM Iterations		25	27	31	20	33
g_ϕ (Roll)	Calibrated	-0.53	-0.68	-0.57	0.97	-0.26
	Std. Dev.	0.070	0.063	0.059	0.046	0.032
	Initial	0	0	0	0	0
g_θ (Pitch)	Calibrated	45.74	44.94	45.45	47.52	46.72
	Std. Dev.	0.157	0.105	0.147	0.056	0.058
	Initial	45	45	45	45	45
g_ψ (Yaw)	Calibrated	3.50	3.06	3.29	5.79	3.92
	Std. Dev.	0.125	0.109	0.118	0.042	0.061
	Initial	0	0	0	0	0
g_x	Calibrated	796.21	801.86	797.87	820.93	781.70
	Std. Dev.	3.001	2.045	2.697	1.204	0.808
	Initial	800	800	800	800	800
g_y	Calibrated	616.68	621.44	618.52	620.38	625.11
	Std. Dev.	2.864	2.525	2.302	1.232	0.857
	Initial	600	600	600	600	600
g_f	Calibrated	-1111.78	-1107.68	-1111.56	-1149.14	-1102.34
	Std. Dev.	2.451	2.166	2.278	0.509	0.805
	Initial	-1100	-1100	-1100	-1100	-1100
$g_{\rho 1}$	Calibrated	0.004	0.011	0.009	-0.022	-0.006
	Std. Dev.	0.0031	0.0025	0.0028	0.0028	0.0014
	Initial	0	0	0	0	0
$g_{\rho 2}$	Calibrated	0.007	-0.006	0.001	-0.036	0.014
	Std. Dev.	0.0048	0.0027	0.0047	0.0050	0.0022
	Initial	0	0	0	0	0

Table 17. Forward Camera, Holding Pattern Maneuver Results

Test Point		2-3	2-7	2-36
Altitude (ft AGL)		5000	5000	10000
Airspeed (KIAS)		150	150	150
SURF Threshold		0.0004	0.0006	0.0006
SURF Octaves		5	5	5
SURF Step		3	3	3
Number of Images		693	461	649
Number of Features		60	60	60
LM Iteration Time		266.2	157.0	450.5
LM Iterations		24	22	55
g_ϕ (Roll)	Calibrated	0.38	1.42	-2.83
	Std. Dev.	0.042	0.084	0.043
	Initial	0	0	0
g_θ (Pitch)	Calibrated	45.62	46.17	42.99
	Std. Dev.	0.043	0.109	0.050
	Initial	45	45	45
g_ψ (Yaw)	Calibrated	4.43	2.86	5.56
	Std. Dev.	0.035	0.096	0.035
	Initial	0	0	0
g_x	Calibrated	806.14	747.08	889.10
	Std. Dev.	0.922	2.223	1.079
	Initial	800	800	800
g_y	Calibrated	643.92	661.82	720.83
	Std. Dev.	0.907	2.378	0.901
	Initial	600	600	600
g_f	Calibrated	-1100.18	-1088.21	-1074.02
	Std. Dev.	0.474	1.600	0.613
	Initial	-1100	-1100	-1100
$g_{\rho 1}$	Calibrated	0.015	-0.079	0.112
	Std. Dev.	0.0015	0.0034	0.0031
	Initial	0	0	0
$g_{\rho 2}$	Calibrated	-0.032	0.072	-0.186
	Std. Dev.	0.0021	0.0050	0.0047
	Initial	0	0	0

Table 18. Forward Camera, Straight Maneuver Results

Test Point		2-03_st	2-07_st	2-27_st	2-36_st
Altitude (ft AGL)		5000	5000	5000	10000
Airspeed (KIAS)		150	150	150	150
SURF Threshold		0.0004	0.0006	0.0006	0.0002
SURF Octaves		5	5	5	5
SURF Step		3	3	3	3
Number of Images		91	86	91	81
Number of Features		60	60	60	60
LM Iteration Time		399.3	164.7	197.1	524.0
LM Iterations		58	64	44	78
g_ϕ (Roll)	Calibrated	2.99	5.37	-1.06	11.39
	Std. Dev.	0.531	1.435	1.027	0.512
	Initial	0	0	0	0
g_θ (Pitch)	Calibrated	39.81	37.80	45.62	42.69
	Std. Dev.	0.641	0.532	0.580	0.270
	Initial	45	45	45	45
g_ψ (Yaw)	Calibrated	8.33	7.37	3.00	10.66
	Std. Dev.	0.310	0.911	0.741	0.313
	Initial	0	0	0	0
g_x	Calibrated	870.22	795.19	797.87	780.52
	Std. Dev.	12.061	19.103	15.812	8.195
	Initial	800	800	800	800
g_y	Calibrated	743.63	853.35	611.76	881.89
	Std. Dev.	9.014	10.472	9.386	8.564
	Initial	600	600	600	600
g_f	Calibrated	-1021.30	-1041.46	-1099.82	-1251.17
	Std. Dev.	11.838	12.725	8.205	7.079
	Initial	-1100	-1100	-1100	-1100
$g_{\rho 1}$	Calibrated	0.088	-0.062	0.001	0.011
	Std. Dev.	0.0142	0.0128	0.0086	0.0073
	Initial	0	0	0	0
$g_{\rho 2}$	Calibrated	-0.115	0.059	0.003	-0.058
	Std. Dev.	0.0143	0.0209	0.0077	0.0100
	Initial	0	0	0	0

Table 19. Forward Camera, S-Turn Maneuver Results

Test Point		2-4	2-25	2-26	2-30	2-31
Altitude (ft AGL)		5000	5000	5000	10000	10000
Airspeed (KIAS)		150	150	150	150	150
SURF Threshold		0.0006	0.0006	0.0006	0.0006	0.0006
SURF Octaves		5	5	5	5	5
SURF Step		3	3	3	3	3
Number of Images		242	260	259	267	257
Number of Features		60	60	60	60	60
LM Iteration Time		157.1	216.9	81.7	161.5	216.3
LM Iterations		16	35	16	32	31
g_ϕ (Roll)	Calibrated	1.23	-0.51	-0.33	4.14	-1.20
	Std. Dev.	0.033	0.029	0.074	0.066	0.039
	Initial	0	0	0	0	0
g_θ (Pitch)	Calibrated	48.35	45.62	45.71	47.25	44.88
	Std. Dev.	0.036	0.030	0.064	0.071	0.039
	Initial	45	45	45	45	45
g_ψ (Yaw)	Calibrated	5.72	3.20	3.15	4.02	5.41
	Std. Dev.	0.025	0.020	0.053	0.050	0.033
	Initial	0	0	0	0	0
g_x	Calibrated	862.65	795.06	789.65	683.50	848.99
	Std. Dev.	0.643	0.715	1.431	1.454	0.804
	Initial	800	800	800	800	800
g_y	Calibrated	582.77	618.57	615.26	664.01	696.95
	Std. Dev.	0.558	0.713	1.043	1.106	0.824
	Initial	600	600	600	600	600
g_f	Calibrated	-1113.54	-1106.61	-1109.05	-1135.99	-1068.02
	Std. Dev.	0.327	0.389	0.679	0.558	0.573
	Initial	-1100	-1100	-1100	-1100	-1100
$g_{\rho 1}$	Calibrated	0.043	0.002	0.002	-0.054	-0.001
	Std. Dev.	0.0019	0.0019	0.0037	0.0029	0.0019
	Initial	0	0	0	0	0
$g_{\rho 2}$	Calibrated	-0.071	0.002	0.005	-0.020	-0.016
	Std. Dev.	0.0029	0.0030	0.0063	0.0042	0.0024
	Initial	0	0	0	0	0

Appendix B – ProfGen Profiles

The following input scripts were used to generate the ProfGen trajectory data used in this thesis. Constant initial conditions were used for each trajectory. Each maneuver shown in Figure 55 was run using a separate configuration file, they are shown together here for conciseness.

```
! State initial conditions
! -----------------------------------------------------------------
-----------
               ! Tstart  DP    initial time of trajectory        sec
0.D0
VsO=90         ! VsO     DP    initial speed                     m/s
0.D0
               ! ROLLO   DP    initial roll angle                deg
0.D0
               ! PITCHO  DP    initial pitch angle               deg
0.D0
HEADO=90       ! HEADO   DP    initial ground path heading       deg
0.D0
               ! ALPHAO  DP    initial wander angle              deg
0.D0
GLATO=35.15    ! GLATO   DP    initial geodetic latitude        deg
0.D0
GLONO=-117.85  ! GLONO   DP    initial geodetic longitude       deg
0.D0
CLONO=-117.85  ! CLONO   DP    initial celestial longitude      deg
0.D0
ALTO=3000      ! ALTO    DP    initial altitude                  m
```

Figure 54. ProfGen Initial Conditions Code

```
! MANUVR SEGLEN   PACC      TACC     DELHED DELPIT  DELROL  PO_DT    FO_DT
!   -      sec     G         G        deg   deg-sec  deg     sec      se
!
========================================================================
========

'Turn Trajectory - 30 deg bank rate turn for 360 deg'
'h'      102.   0.       0.575    360.     0.      0.      55.      0.25


------------------------------------------------------------------------
'Climbing Turn - 11 deg climb, 30 deg bank rate turn for 360 deg''
'vert'     3   0.       1.2      0.      28.      0.      55.      0.25
'h'      120.   0.       0.575    360     0.      0.      55.      0.25


------------------------------------------------------------------------


'Hold Pattern - straight for 30 sec, 30 deg bank 180, repeat'
's'      30.    0.       0.       0.      0.      0.      55.      0.25

'h'      60.    0.       0.575    180.    0.      30.     55.      0.25

's'      30.    0.       0.       0.      0.      0.      55.      0.25

'h'      60.    0.       0.575    180.    0.      30.     55.      0.25
------------------------------------------------------------------------
'Straight - Straight for 30 sec'
's'      30.    0.       0.       0.      0.      0.      55.      0.25


------------------------------------------------------------------------


'S-Turn - Standard rate turn for 90 deg, reverse'
'h'      30.    0.       0.575    90      0.      0.      55.      0.25
'h'      30.    0.       0.575    -90     0.      0.      55.      0.25
```

Figure 55. ProfGen Maneuver Configuration Code

Bibliography

[1] Curtis Cohenour and Frank van Graas, "Image Georegistration," Avionics Engineering Center, Ohio University, Athens, Ohio, 2011.

[2] Philip Lorenzini et al., "Automatic Calibration of an Aerial Camera (Project "Have SURF")," USAF Test Pilot School, Edwards AFB, Technical Information Memorandum 2012.

[3] Michael J. Veth, "Optical and Inertial Sensor Fusion," Air Force Institute of Technology, Wright Patterson AFB, PhD Dissertation 2006.

[4] Michael B. Nielson, "Development and Flight Test of a Robust Optical-Inertial Navigation System Using Low-Cost Sensors," Air Force Institute of Technology, Wight Patterson AFB, MS Thesis 2008.

[5] Curtis Cohenour and Frank van Grass, "Ohio Georegistration Results," Ohio University, Athens, Presentation Slides 2011.

[6] Q. Luong and O. Faugeras, "Self_Calibration of a Moving Camera from Point Correspondences and Fundamental Matrices," *International Journal of Computer Vision*, vol. 22, no. 3, pp. 261-289, 1997.

[7] Apostolos Mamatas, Ahmed Mohamed, and Andrey Soloviev, "Using Inertial Measurements to Aid in Camera Intrinsic Parameter Estimation," in *Interational Techinical Meeting of The Institute of Navigation*, San Diego, 2011.

[8] Michael G. Giebner, "Tightly-Coupled Image-Aided Inertial Navigation System via a Kalman Filter," Air Forct Institute of Technology, Wright Patterson AFB, MS Thesis 2003.

[9] Zhang Zhengyou, "A Flexible New Technique For Camera Calibration," *IEEE Transactions on Pattern Analysis amd Machine Intelligence*, vol. 22, no. 11, November 2000.

[10] David A. Forsyth and Jean Ponce, *Computer Vision: A Modern Approach*. Upper Saddle River, New Jersey: Prentice Hall, 2003.

[11] Richard Hartley and Andrew Zisserman, *Multiple View Geometery in Computer Vision*, 2nd ed. Cambridge, United Kingdom: Cambridge University Press, 2003.

[12] Richard Hartley, "In Defense of the Eight-Point Algorithm," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 19, no. 6, pp. 580-593, June 1997.

[13] K. Madsen, H.B. Nielsen, and O. Tingleff, "Methods for Non-Linear Least Squares Problems," Technical University of Denmark, 2004.

[14] Manolis I. A. Lourakis and Antonis A. Argyros, "SBA: A Software Package for Generic Sparse Bundle Adjsutment," *ACM Trans. Math. Softw.*, vol. 36, no. 1, March 2009.

[15] Jean-Yves Bouguet. (2010, July) Camera Calibration Toolbox for Matlab. [Online]. http://www.vision.caltech.edu/bouguetj/calib_doc/

[16] K.H. Strobel, W. Sepp, S. Fuchs, C. Paredes, and K. Arbter. (2011) DLR CalDe and DLR CalLab. [Online]. http://www.robotic.dlr.de/callab/

[17] Christopher Evans. (2011, November) The OpenSURF Computer Vision Library. [Online]. http://www.chrisevansdev.com/computer-vision-opensurf.html

[18] U.S. Geological Survey. (2013, January) Earth Explorer. [Online]. http://earthexplorer.usgs.gov/

[19] Stanton H. Musick, "User's Guide for ProfGen, A Trajectory Generator," Air Force Reasearch Lab, Sensors Directorate, Wright-Patterson AFB, User Manual 2004.

[20] Novatel Inc. (2012, July) NovaTel. [Online]. http://www.novatel.com/assets/Documents/Papers/HG1700_SPAN58.pdf

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From – To)*
21-03-2013	Master's Thesis	Aug 2010 – Mar 2013

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Sensitivity Analysis of an Automated Calibration Routine for Airborne Cameras	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
Lorenzini, Philip, E., Captain, USAF	5e. TASK NUMBER
	5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAMES(S) AND ADDRESS(S)	8. PERFORMING ORGANIZATION REPORT NUMBER
Air Force Institute of Technology Graduate School of Engineering and Management (AFIT/EN) 2950 Hobson Way, Building 640 WPAFB OH 45433-7765	AFIT-ENG-13-M-51

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
Intentionally left blank	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

13. SUPPLEMENTARY NOTES
This material is declared a work of the U.S. Government and is not subject to copyright protection in the United States.

14. ABSTRACT
Given a known aircraft location, a set of camera calibration parameters can be used to correlate features in an image with ground locations. Previously, these calibration parameters were obtained during preflight with a lengthy calibration process. A method to automate this calibration using images taken with an aircraft mounted camera and position and attitude data is developed. This thesis seeks to determine a partial set of circumstances that affect the accuracy of the calibration results through simulation and experimental flight test.

A software simulator is developed in which to test an array of aircraft maneuvers, camera orientations, and noise injection. Results from the simulation are used to prepare test points for an experiment flight test conducted to validate the calibration algorithm and the simulator.

Real world flight test methodology and results are discussed. Images of the ground along with precise aircraft navigation and time data were gathered and processed for several representative aircraft maneuvers using two camera orientations.

15. SUBJECT TERMS
SURF, Speeded Up Robust Features, Aerial Camera Calibration, Automatic Camera Calibration, Robust Feature Detection, Feature Calibration

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Raquet, John, PhD. USAF ADVISOR
			UU	125	19b. TELEPHONE NUMBER *(Include area code)*
U	U	U			(937) 255-3636 x4580 (john.raquet@afit.edu)

Standard Form 298
(Rev. 8-98)
Prescribed by ANSI Std. Z39-18