DETERMINATION OF STRUCTURE FROM MOTION USING AERIAL IMAGERY

THESIS

Paul R. Graham, First Lieutenant, USAF

AFIT/GCS/ENG/05-06

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

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

AFIT/GCS/ENG/05-06

DETERMINATION OF STRUCTURE FROM MOTION USING AERIAL IMAGERY

THESIS

Presented to the Faculty

Department of Electrical and Computer Engineering

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

In Partial Fulfillment of the Requirements for the

Degree of Master of Science

Paul R. Graham, BS

First Lieutenant, USAF

March 2005

AFIT/GCS/ENG/05-06

DETERMINATION OF STRUCTURE FROM MOTION USING AERIAL IMAGERY

Paul R. Graham, BS

First Lieutenant, USAF

Approved:

_/signed/_____	_____
John F. Raquet, PhD (Chairman)	Date
_/signed/_____	_____
Gilbert L. Peterson, PhD (Member)	Date
_/signed/_____	_____
Steven C. Gustafson, PhD (Member)	Date

Acknowledgments

I would like to express my sincere appreciation to my faculty advisor, Dr John Raquet, for his guidance and support throughout the course of this thesis effort. The insight and experience was certainly appreciated. I would, also, like to thank those who took the time to proof read this effort.

Paul R. Graham

Table of Contents

Page

List of Figures

Page

List of Tables

Page

Abstract

The structure from motion process creates three-dimensional models from a sequence of images. Until recently, most research in this field has been restricted to land-based imagery. This research examines the current methods of land-based structure from motion and evaluates their performance for aerial imagery.

Current structure from motion algorithms search the initial image for features to track though the subsequent images. These features are used to create point correspondences between the two images. The correspondences are used to estimate the motion of the camera and then the three-dimensional structure of the scene. This research tests current algorithms using synthetic data for correctness and to characterize the motions necessary to produce accurate models. Two approaches are investigated: full Euclidian reconstructions, where the camera motion is estimated using the correspondences, and navigation-aided Euclidian reconstructions, where the camera motion is calculated using the Global Positioning System and inertial navigation system data from the aircraft.

Both sets algorithms are applied to images collected from an airborne blimp. It is found that full Euclidian reconstructions have two orders of magnitude more error than navigation-aided Euclidian reconstructions when using typical images from airborne cameras.

DETERMINATION OF STRUCTURE FROM MOTION USING AERIAL IMAGERY

I. Introduction

1.1 Background

Computer vision research seeks to develop systems that give computers the ability to "see" in a three-dimensional world. To a computer, images received from a digital camera or scanner are a collection of positive numbers that measure the amount of light reflected from a particular location [8]. It is desirable to turn this measurement of light from two or more images into a three-dimensional representation of the scene. Creating these three-dimensional models from two-dimensional images is known as "structure from motion".

Researchers have successfully developed a structure from motion pipeline system that uses images gathered from land-based cameras. Land-based cameras produce pictures from a stationary point on the ground. These images normally have a higher resolution than images taken from aircraft, and result in the production of a higher quality model.

This research expands the existing structure from motion research into the realm of aerial imagery. Most previous structure from motion research up to this point has involved land-based camera images. The research reported here seeks to produce a pipeline system that creates models from aerial images and to evaluate the potential of structure from motion using airborne imagery. Aerial images differ from land images

due to the distance and point of view from which they are taken. These properties decrease the quality of the image and increase the probability of an error in the resulting model.

1.2 Problem Statement and Focus

The focus of this research is to implement a structure from motion algorithm for airborne imagery and to evaluate its performance.

The thesis implements and documents a structure from motion application using available methods for each step in the structure from motion pipeline. The application is designed to determine the capabilities of the structure from motion pipeline for aerial imagery, and the thresholds that produce the most accurate results.

1.3 Investigative Questions

This thesis seeks to answer the following questions:

- Can structure from motion be accomplished using aerial imagery?
- If so, what factors (number or images, motion of the camera, navigation information, etc.) are most important for obtaining a model from aerial imagery using a structure from motion pipeline?

The first question involves developing a system that implements structure from motion modules. The second question involves developing and documenting a method for comparing three-dimensional models to a baseline model.

1.4 Methodology

There are two main tasks to this research effort. The first task develops a system that is capable of producing three-dimensional models from the various components of the pipeline. The second task devises a method for comparing the output models to determine the most accurate methods and thresholds.

To achieve the first task, a structure from motion application is created. Since there are numerous methods for each of the pipeline steps in the process, the application accomplishes a three-dimensional model reconstruction in a modular manner. It takes the intrinsic camera parameters and images as inputs and generates three-dimensional structure accordingly. This application is then used with test data to create models from real aerial imagery.

The second task consists of two steps, simulation tests and flight tests. Simulation testing involves creating a three-dimensional model. Then the model is used to render a sequence of images as inputs. Next these synthetic images are applied to the system, and the resulting model is compared to the original. Flight testing involves using real aerial images to create the three-dimensional model. Aerial imagery is provided by the Air Force Research Laboratory Munitions Directorate

1.5 Assumptions/Limitations

The location of the camera in relation to the Global Positioning System (GPS) receiver on the aircraft is assumed to be known. Since the aircraft and the camera behave like rigid-body structures, the camera position can be inferred from the coordinates transmitted by the GPS receiver. The pointing direction of the camera with respect to the

Internal Navigation System (INS) is also assumed to be known. These two assumptions allow the user to know the location and pointing direction of the camera.

This research makes the assumption that the intrinsic camera parameters are already known or that there is sufficient information to calculate them from the images before executing the algorithm (see camera calibration methods in Chapter II). The camera parameters maybe calculated from the images themselves, but this possibility is left for future research efforts.

This research also assumes that all objects in the scene have the Lambertian property. Materials with the Lambertian property do not change appearance when the viewing location changes [8]. This assumption simplifies the detection and tracking of features, which is a crucial step in the pipeline.

II. Literature Review

2.1 Chapter Overview

This chapter describes existing structure from motion methods. First, the camera model and calibration methods are introduced along with their supporting functions. Second, the pipeline is discussed generically, and competing methods are described in the pipeline order. Finally, other research in the field of structure from motion field is discussed along with how it is complemented by the research reported here.

2.2 Camera Model

The model for tracing points in space to pixels in an image must account for the following transformations [8]:

- Coordinate transformation from the real-world frame to the camera frame
- Projection of a three-dimensional coordinate space onto a two dimensional coordinate plane
- Transformations between different possible choices of image coordinate frames

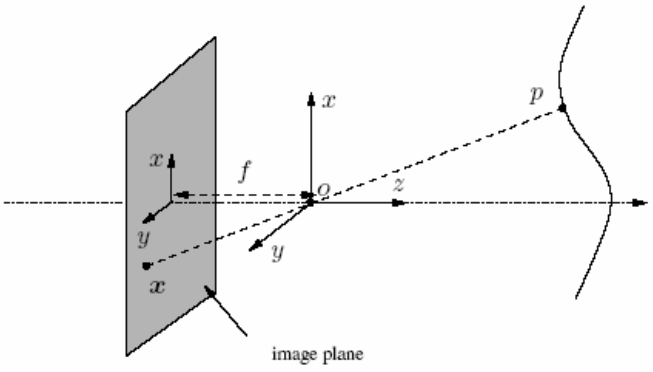


Figure 2-1 – Image Formation [8]

The world frame is a three dimensional coordinate system with respect to some arbitrary origin. The camera frame is also a three-dimensional coordinate system; however, the origin is considered to be the location of the optical center of the lens. The transformation between the world frame and the camera frame is governed by a rigid-body transformation and is modeled as

$$X = RX_O + T, \tag{2-1}$$

where X_O is the point with respect to the world reference frame, X is the point with respect to the camera frame, R is the direction the coordinate system must be rotated to match the direction the camera is pointed, and T is the translation vector between the origins of the camera frame and the world frame.

Projecting the three-dimensional coordinate space onto the two-dimensional image plane is accomplished using

$$x = \begin{bmatrix} \tilde{x} \\ \tilde{y} \end{bmatrix} = \frac{f}{Z}\begin{bmatrix} X \\ Y \end{bmatrix}, \tag{2-2}$$

where $\tilde{x}$ and $\tilde{y}$ are the camera plane coordinates, f is the distance from the camera origin, O, to the image plane (Figure 2-1), and X, Y and Z are the three-dimensional coordinates with respect to the camera frame. This equation can be expressed in homogeneous coordinates

$$Z\begin{bmatrix} \tilde{x} \\ \tilde{y} \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}\begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}. \tag{2-3}$$

Adjusting Eq. 2-3 for the physical size of a pixel in the *x* and *y* directions, the skew factor of each pixel and the optical center of the camera on the image plane yields

$$Z\begin{bmatrix}\tilde{x}\\ \tilde{y}\\ 1\end{bmatrix}=\begin{bmatrix}fs_x & s_\theta & o_x\\ 0 & fs_y & o_y\\ 0 & 0 & 1\end{bmatrix}\begin{bmatrix}1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 1 & 0\end{bmatrix}\begin{bmatrix}X\\ Y\\ Z\\ 1\end{bmatrix}, \tag{2-4}$$

where s_x and s_y are the dimensions of the pixels, s_θ is the skew factor of the pixel, and o_x and o_y are the coordinates of the optical center on the image plane. These values are the intrinsic parameters of the camera and account for the third transformation in the camera model. The matrix that includes these values is the camera calibration matrix

$$K=\begin{bmatrix}fs_x & s_\theta & o_x\\ 0 & fs_y & o_y\\ 0 & 0 & 1\end{bmatrix}. \tag{2-5}$$

When all three transformations are combined, the camera is modeled by

$$Z\begin{bmatrix}\tilde{x}\\ \tilde{y}\\ 1\end{bmatrix}=\begin{bmatrix}fs_x & s_\theta & o_x\\ 0 & fs_y & o_y\\ 0 & 0 & 1\end{bmatrix}\begin{bmatrix}1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 1 & 0\end{bmatrix}\begin{bmatrix}R & T\\ 0 & 1\end{bmatrix}\begin{bmatrix}X_O\\ Y_O\\ Z_O\\ 1\end{bmatrix}. \tag{2-6}$$

2.3 Camera Calibration Methods

The intrinsic parameters of the camera provide crucial information for the structure from motion pipeline. A method for uncovering these camera parameters is described in [10]. The process requires knowledge of the actual location of some points of interest

that appear in the images with respect to a known camera location. It also requires knowledge of the pointing direction of the camera with respect to the inertial navigation system output of the airplane. According to [10], the location of the interest points and the location of the camera can be discovered using GPS data. Once the locations of the points and the camera are known, the camera calibration algorithm creates a vector from the camera to each of the points of interest.

Using several of these vectors, the location of the camera, the inertial navigation system data, and the *x* and *y* pixel coordinates of the known points of interest, the five internal camera parameters (*K*) and the three camera angle mounting errors are estimated using a gradient search method. This method takes advantage of the camera model described earlier (see Eq. 2-6). An initial guess for the unknown camera parameters is made. According to [8], the initial estimation for the camera parameters typically is

- fs_x = fs_y = number of pixels in the *x* dimension times a variable from the interval [0.5, 2]
- s_θ = 0 or 1
- o_x = the number of pixels in the image in the *x* dimension
- o_y = the number of pixels in the image in the *y* dimension

This estimate of the intrinsic camera parameters allows the calculation of pixel coordinates for each of the points of interest. The calculated pixel coordinates are then compared to the corresponding known pixel coordinates. Finally, the estimated

parameters are adjusted and the process is repeated until the calculated pixel coordinates converge to the actual pixel coordinates.

2.4 Structure from Motion Pipeline

Taking images and extracting the three-dimensional scenes they represent is accomplished using a pipeline architecture (Figure 2-2). This structure consists of modules that take an input from the user or prior module and produce outputs to drive the next module or the final model. Some of the modules have a number of associated algorithms. The methods used in this research are based on the methods described in [8].

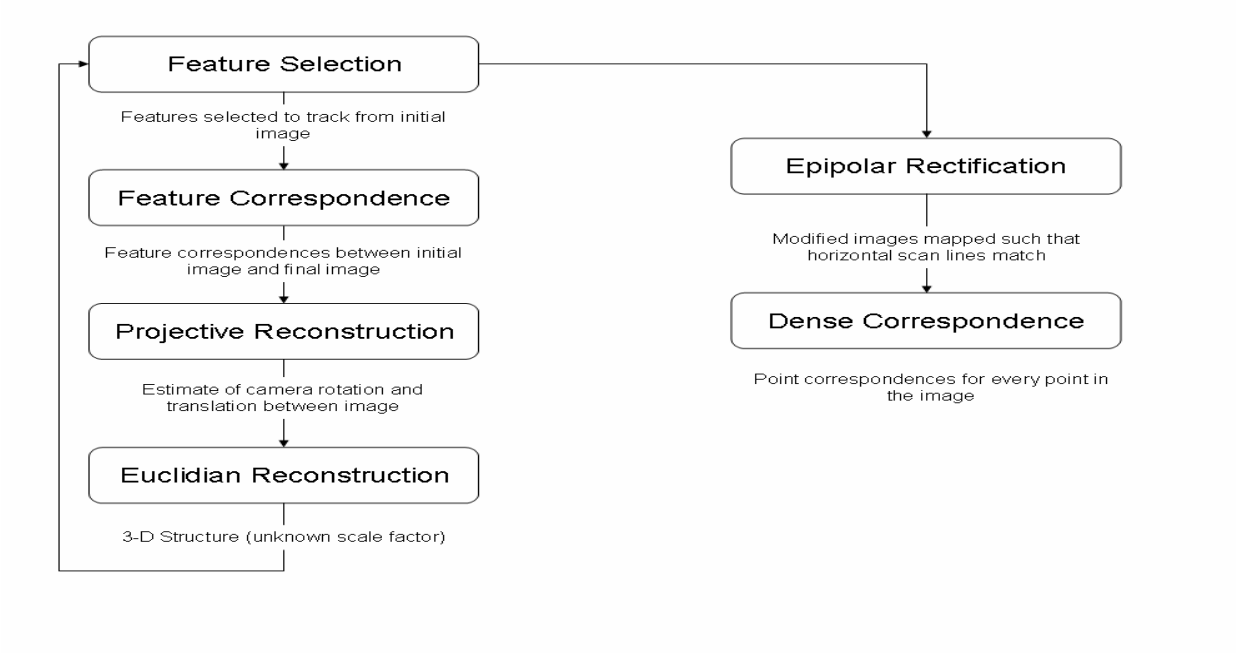


Figure 2-2 – Structure from Motion Pipeline

2.4.1 Pipeline Descriptions

Feature selection is the first step in the structure from motion pipeline. It is closely entwined with the second step, feature correspondence. It is also one of the most

important steps because selecting poor features causes the pipeline to fail. Feature selection takes the initial image in the image sequence as input and finds a set of features that have the qualities necessary to track through the subsequent images.

There are two conflicting goals involved in selecting features. First, the selected features should be distributed throughout the image [8]. To accomplish this distribution, the image is split into tiles of equal size and features are selected within each tile. Second, each feature must be separated by a distance from other features; otherwise they may be construed as the same feature. This separation is accomplished by selecting the single most prominent feature within the search window.

Feature correspondence is the second step in the structure from motion pipeline. It takes the features found in the original image as input and determines the displacement of these features in the subsequent images. The threshold defined in this algorithm determines the number the features kept though the tracking process. This portion of the pipeline is the most important and complicated step in the process [8].

Projective reconstruction is the third step in the structure from motion pipeline. It recovers the three dimensional structure of the scene up to a projective transformation. The result of this process is a weaker form of the three-dimensional structure, as some data on the position of points is lost. The projective reconstruction process can be done with two images, adding additional images one at a time if extra images are available, or multiple images all at once. Projective reconstruction takes sets of correspondences between the images as inputs and outputs the three dimensional structure for each correspondence [8].

The *Euclidian reconstruction* is the "true" world representation of the scene. It contains all the three dimensional information up to a scalar factor of the image. When the intrinsic camera parameters are known, the Euclidian reconstruction problem reduces to a linear triangulation problem [8]. Solving the linear triangulation problem requires knowledge of the location of the camera in both images. The camera movements are calculated from the eight-point algorithm (see 8-point Algorithm in Appendix B). These reconstruction methods take the internal camera parameters and image correspondences as inputs and produce the true Euclidian structure of the images.

With no knowledge of the intrinsic camera parameters, a linear transformation H relates the Euclidian structure to its corresponding projection structure. The goal of the Euclidian upgrade from the projective reconstruction is to calculate H. Euclidian upgrade methods take the projective transformations and the feature correspondences found between images as inputs and produce the true three-dimensional structure for the correspondences.

Epipolar rectification and *dense matching* are the final steps in creating the three-dimensional model. Epipolar rectification entails finding two linear transformations of the projective coordinates that transform each image so that its epipole is at infinity in the x-axis direction [8]. This process warps the images in such a manner that all the pixels along a scan line in the first image correspond to pixels along the same scan line in the second image. So, modifying the images reduces the amount of searching necessary to track features across the images to just one dimension. At this stage, most of the pixels are matched in each image, and dense correspondence can be accomplished using the

approaches described earlier. Then projective and Euclidian reconstructions can be accomplished en masse.

2.4.2 Structure from Motion Algorithm Descriptions

Feature Selection - Harris Corner Detector

The Harris Corner Detector is one of the most straightforward methods to extract features. It limits the type of features detected to point features, which simplifies the process. The Harris Detector selects a point when its quality meets the Harris Criterion,

$$C(x) = \det(G) + k \times trace(G), \tag{2-7}$$

computed over a window region of the image. In this equation k is a constant chosen by the designer and G is the 2x2 matrix

$$G = \begin{bmatrix} \sum_{W(x)} I_x{}^2 & \sum_{W(x)} I_x I_y \\ \sum_{W(x)} I_x I_y & \sum_{W(x)} I_y{}^2 \end{bmatrix} \tag{2-8}$$

where I_x and I_y are the gradients obtained by convolving the image I with the derivatives of a pair of Gaussian filters. If $C(x)$ exceeds some user defined threshold, it is selected as a feature [3].

Feature Tracking

Features are tracked by determining d, the displacement of a feature x between two images. Other research has shown that d can be found using,

$$d = -G^{-1}b \tag{2-9}$$

where G is the same matrix used to determine the Harris criterion and b is

$$b = \begin{bmatrix} \sum_{W(x)} I_x I_t \\ \sum_{W(x)} I_y I_t \end{bmatrix}, \tag{2-10}$$

where $I_t \doteq I_2 - I_1$ is the difference between the two views. Yi Ma, et al. describe a robust algorithm that implements this tracking feature [8]. Their implementation is a layered approach. The original image is down-sampled by a factor of two until several layers of coarseness are available. Starting with the coarsest image, d_i is calculated. The displacement is scaled up by a factor of two and the window, *W(x),* around the feature is moved to $W(x+2d_i)$. Then, using the new window, d_{i-1} is calculated for the next coarsest image. This process continues until full resolution is obtained. Finally, the total displacement is found by summing the interim displacements multiplied by their scaled factor

$$d = \sum_{i=1}^{k} 2^{i-1} d_i \tag{2-11}$$

Projective Reconstruction: Two Views

In [8], the authors begin the projective reconstruction by guessing the calibration matrix *K*. This step typically involves choosing the optical center, assuming the pixels are square, and estimating the focal length. The normalizing transformation *H* is substituted in the eight-point algorithm with *K* to estimate the fundamental matrix *F* [8]. The epipole *T'* is then computed as the null space of F^T. Then, *v* and v_4 are chosen so that the rotational portion *F* is as close as possible to a small rotation. Selecting the first image as the reference image, the projection matrices are

$$\prod_{ip} = \begin{bmatrix} I & 0 \end{bmatrix} \quad \prod_{ip} = \begin{bmatrix} v_4 \left(\hat{T}'\right)^T F + T' v^T & T' \end{bmatrix} = \begin{bmatrix} R & T' \end{bmatrix}. \tag{2-12}$$

If the projection matrices are written in terms of their three row vectors, the unknown structure satisfies

$$\begin{matrix} \left(x_1 \pi_1^{3T} - \pi_1^{1T}\right) X_p = 0, & \left(x_1 \pi_1^{3T} - \pi_1^{2T}\right) X_p = 0 \\ \left(x_2 \pi_2^{3T} - \pi_2^{1T}\right) X_p = 0, & \left(x_2 \pi_2^{3T} - \pi_2^{2T}\right) X_p = 0 \end{matrix}, \tag{2-13}$$

where π_i^j is the j^{th} row vector in the projection matrix for i^{th} image and x_i and y_i are the pixel coordinates in the i^{th} image of the feature. Writing the projection matrices this way reduces the problem of finding three dimensional structure to finding a least squares solution of a linear system of equations $MX_p=0$. The solution for each point is given by the eigenvector M^TM that corresponds to the smallest eigenvalue.

Euclidian Reconstruction

When the eight-point algorithm is executed using the true intrinsic camera parameters instead of an estimation of the camera parameters, the triangulation method used to determine the projective structure provides the Euclidian structure instead (for details reference Projective Reconstruction: Two Views).

Simple Epipolar Rectification

The first step in epipolar rectification computes the fundamental matrix F. From this matrix, the epipole e_2 is found by determining the right null space of F, and H_2 is computed using

$$H_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -1/x_e & 0 & 1 \end{bmatrix} \begin{bmatrix} x_e \\ 0 \\ 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & -o_x \\ 0 & 1 & -o_y \\ 0 & 0 & 1 \end{bmatrix}. \tag{2-14}$$

A linear transformation, H, is then selected where $\hat{T}'H \sim F$ and

$$H_1 = H_2 H \ . \tag{2-15}$$

Finally, all the image coordinates are transformed using $x_1=H_1x'_1$ and $x_2=H_2x'_2$, and the z-coordinate is normalized to one by interpolating the intensity values for coordinates outside the pixel grid [8].

2.5 Other Structure from Motion Research

Other research efforts have developed different methods for completing each of the pipeline steps. This section discusses these methods.

According to [4] and [9] feature detection can include line and edge features as well as point features. These types of detection algorithms run faster than the point detection but produce less reliable results.

In [1], the author describes how to track features in widely separated views. To produce a model that is closer to the true Euclidian structure, larger motions are required. However, this research focuses on images procured from video. It is the nature of images acquired this way to have a small amount of motion between them. To accommodate this effect, feature tracking can be done between sequential images and then the correspondences from the two images that are most widely separated can be used.

There are several different methods for recovering the projective reconstruction and the Euclidian reconstruction described in [2] and [5]. These methods accommodate changing and unknown camera parameters and are beyond the scope of this research because here it is assumed that the camera parameters are known.

In [6], the author describes a two-camera approach to reconstructing the three-dimensional shape from images. A two camera approach involves two static cameras. This approach is not applicable to the research reported here because aerial imagery is captured using a single moving camera.

2.6 Summary

This chapter describes the structure from motion architecture and how it creates a pipeline that transforms images into three-dimensional models. The modules and their interaction within the pipeline are discussed and different implementations of the modules are described. Finally, some ways to discover the camera calibration parameters necessary to run images through the structure from motion pipeline are presented.

III. Methodology

3.1 Chapter Overview

This chapter describes the methods used for implementing the structure from motion algorithm and the tests conducted to confirm its performance. It also discusses how the camera for the flight tests was calibrated.

3.2 Algorithm Implementation

3.2.1 Camera Calibration

The intrinsic camera parameters are estimated using the camera calibration algorithm from Chapter II. The Global Positioning System (GPS) locations for several points of interest are captured from the video and used in the flight tests (see Figure 3-1 and Appendix B). The GPS positions and inertial navigation system (INS) attitudes of the camera for five images in which the points of interest appear are also recorded (see Table 3-1). The actual pixel coordinates for each of the points of interest are recorded using visual inspection in each image in which they appear (see Figure 3-2). Ninety-eight of these correspondences are created to ensure that the camera calibration matrix is over determined and to reduce the impact of errors in determining the exact pixel coordinates. The results of the algorithm are reported in Chapter IV.

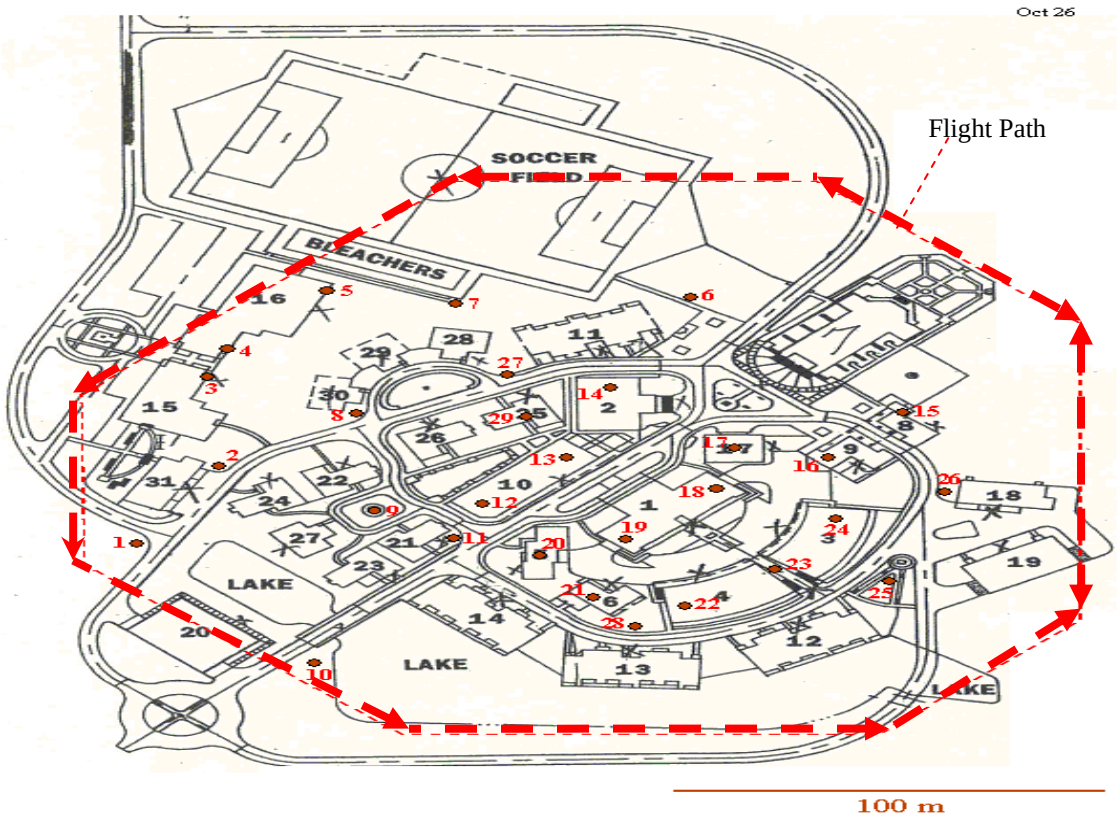


Figure 3-1 – Map of Test Area

Table 3-1 – Image Timestamp and Location Data for Camera Calibration Algorithm

Image Number	1	2	3	4	5
Timestamp (Day:Hr:Min:Sec:mS)	304:15:05:19:532	304:15:05:20:533	304:15:05:22:869	304:15:05:24:437	304:15:05:25:905
Easting from UTM 18 (m)	293905.31	293909.10	293917.19	293922.23	293926.60
Northing from UTM 18 (m)	3838302.84	3838306.53	3838315.28	3838321.50	3838327.64
HAE (m)	149.43	149.90	150.93	151.60	152.21
Roll (deg)	1.4245	1.5558	1.7959	2.2351	2.6132
Pitch (deg)	-51.865	-51.992	-51.414	-50.805	-50.302
Yaw (deg)	217.01	216.13	213.91	212.15	210.31

Figure 3-2 – Example Image used in Camera Calibration

3.2.2 Structure from Motion Implementation

The structure from motion algorithm implemented is modified from the pipeline description in Chapter II. By assuming that the intrinsic parameters of the camera are known, portions of the projective reconstruction step of the pipeline can be skipped. Also, reconstructing a complete scene using the dense correspondence methods is beyond the scope of this research. Figure 3-3 shows the steps of the pipeline again, with the steps that are not implemented. These algorithms are implemented as described in Chapter II.

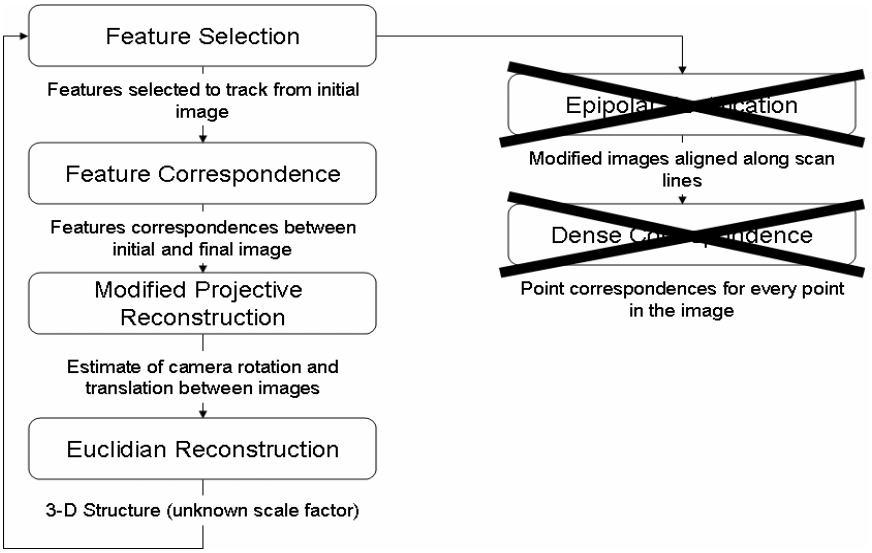


Figure 3-3 – Modified Pipeline for Full Euclidian Reconstruction

The reconstruction algorithms are also implemented to incorporate the navigation information associated with aircraft. Since the camera is attached to an aircraft, its true position and rotation can be calculated from GPS and INS data. This information is then inserted into the Euclidian reconstruction step in place of the estimated rotations and translations. Figure 3-4 shows the pipeline after these modifications are accomplished.

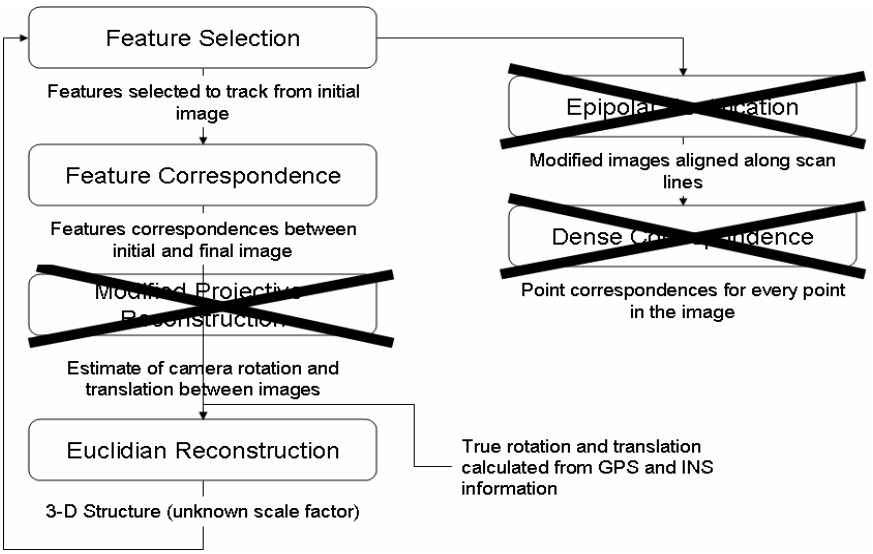


Figure 3-4 – Modified Pipeline for Research Navigation Aided Euclidian Reconstruction

The feature selection process has several static parameters (see Table 3-2). The values used in the code for these parameters are those recommended by [8]. The window for selection, the threshold for rejecting a feature, the minimum distance between features, and the boundary threshold are the most crucial. The window for selection is the window around the point used to determine the strength of the feature. This parameter solves the aperture problem of the selection process. Since it would be computationally hard to look at the entire image at once, the algorithm must look at windows around each point to find features that can be tracked. If the window around the points is decreased too much, the quality of the selected features deteriorates. The threshold for determining if a feature is of high enough quality to be tracked is the second parameter. A lower value for this parameter finds more features of lesser quality, while a higher value finds fewer features of greater quality. The third parameter necessary for the feature selection process is the distance between feature points. This parameter ensures that the features tracked by the tracking algorithm are the initial features found by the selection algorithm. The final parameter, the boundary threshold, ensures that features near the border of image are not selected by excluding them from the search area.

Table 3-2 – Values for Feature Selection and Feature Tracking Static Parameters

Parameter Name	Variable Name in Code	Value Used in Code
Window for Selection	`winx, winy`	1
Distance between Features	`spacing`	5
Rejection Threshold	`thresh`	0.05
Boundary Threshold	`boundary`	100
Re-sampling Pyramid Size	`levelmax`	2

The feature tracking algorithm includes the same static parameters as the feature selection process as well as another feature, the number of levels in the re-sampling pyramid. This parameter describes the number of levels of re-sampling needed to track images through the video. The amount of motion between images determines the number of levels needed to reliably track features. Since this research concerns in images gathered from a video camera, the motion between frames is small, and a lower number of levels can be used (see Table 3-2).

Once the correspondences are established, performing the Euclidian reconstruction is a simple linear process that involves no static parameters.

IV. Tests, Analysis and Results

4.1 Chapter Overview

This chapter describes the simulation tests and flight tests for the implemented structure from motion pipeline. Simulation testing is accomplished by comparing the results that the synthetic images produce when input into the pipeline to the actual model used to create the synthetic images. These tests are done to accomplish two goals. First, they show that the structure from motion pipeline works and second, they discover the motion limitations of the algorithms. The flight tests are conducted to discover the applicability of the pipeline to real aerial imagery.

4.2 Simulation Tests

The feature detection and tracking algorithms were tested using a combination of visual inspection and error metric methods. The following tests were run to investigate the properties of the reconstruction portion of the algorithm.

4.2.1 The Model

To validate the algorithm, a simple three-dimensional model was developed. Twelve points were created to represent a three-dimensional model, and their location with respect to the initial camera was recorded (see Figure 4-1 and Table 4-1). The initial camera was centered 6 units in front of the 6-5-11-12 face of the model aligned with points 3 and 9. The model was designed such that if the results from the structure from motion pipeline differ from the original model by a rotation about one of the axes, this rotation would be detected.

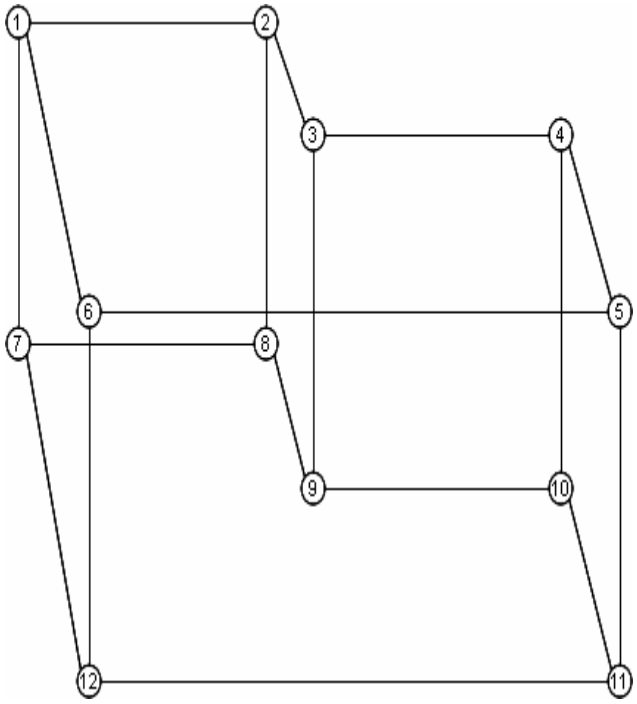


Figure 4-1 – Simple Model for Simulation Testing

Table 4-1 – Points of Interest Locations with Respect to Initial Camera

Point	Offset From Initial Camera Location $[X \quad Y \quad Z]^T$
1	$[-1 \quad 1 \quad -8]^T$
2	$[0 \quad 1 \quad -8]^T$
3	$[0 \quad 1 \quad -7]^T$
4	$[1 \quad 1 \quad -7]^T$
5	$[1 \quad 1 \quad -6]^T$
6	$[-1 \quad 1 \quad -6]^T$
7	$[-1 \quad -1 \quad -8]^T$
8	$[0 \quad -1 \quad -8]^T$
9	$[0 \quad -1 \quad -7]^T$
10	$[1 \quad -1 \quad -7]^T$
11	$[1 \quad -1 \quad -6]^T$
12	$[-1 \quad -1 \quad -6]^T$

4.2.2 The Images

Following the mathematical camera model described in Eq. 2-6, a virtual camera model was created:

$$K_{Virtual} = \begin{bmatrix} 720 & 0 & 360 \\ 0 & 720 & 240 \\ 0 & 0 & 1 \end{bmatrix}. \tag{4-1}$$

Using this virtual camera, the 12 points of the structure were projected onto a virtual image plane, creating images of the model. In addition, the locations where the features of interest appear in each image were recorded. To create the subsequent images for the reconstruction process, two sets of images were created. For the first set, the camera was first translated from its original position to one of the four locations shown in Table 4-2.

Table 4-2 – Translations Used to Validate Algorithm

Case	Translation in the *X* direction	Translation in the *Y* direction	Translation in the *Z* direction
1	1.0	1.0	1.0
2	0.0	1.0	1.0
3	1.0	0.0	1.0
4	1.0	1.0	0.0

Then, for each translation a total of 16 rotations on the camera were performed (all possible combinations of 0°, 30°, 60°, 90° about the *x* and *y* axes). For the second set the camera was translated from its original position by 0.01 units 200 times in the following directions:

- *x* axis only (from $[0 \quad 0 \quad 0]^T$ to $[2 \quad 0 \quad 0]^T$)
- *y* axis only (from $[0 \quad 0 \quad 0]^T$ to $[0 \quad 2 \quad 0]^T$)
- *z* axis only (from $[0 \quad 0 \quad 0]^T$ to $[0 \quad 0 \quad 2]^T$)

- *x* and *y* axis (from $[0 \quad 0 \quad 0]^T$ to $[2 \quad 2 \quad 0]^T$)
- *x* and *z* axis (from $[0 \quad 0 \quad 0]^T$ to $[2 \quad 0 \quad 2]^T$)
- *y* and *z* axis (from $[0 \quad 0 \quad 0]^T$ to $[0 \quad 2 \quad 2]^T$)
- *x, y* and *z* axis (from $[0 \quad 0 \quad 0]^T$ to $[2 \quad 2 \quad 2]^T$)

The subsequent images determine what sorts of motions produce an accurate model. The motions that the camera can undergo span a spectrum from weak motion to rich motion. Weaker motions provide less information to the algorithm for the estimate of the camera movement and produce distorted models. As motions provide more information to estimate the camera movements, they become richer and produce more accurate models.

These image correspondences are then fed into the structure from motion process to capture the effects of different camera movements.

4.2.3 Noise Introduction

The images described above provide exact point correspondences between the original image and the subsequent images. To determine the effect of noise on the reconstruction process, Gaussian noise with a varying standard deviation (from 1 to 20) was added to the image correspondences. These noisy correspondences were then used as inputs to determine the effect of noise on the produced model. The reconstruction process was then repeated and the resulting models were compared to the original.

Recall that the Euclidian reconstruction results in a scale factor that may be different than the real world units. However there is a desire to evaluate the "accuracy" of the results with correspondence errors. In order to make a valid comparison, the scalar

multiplier for the calculated model is found by dividing the actual coordinates by the calculated coordinates. Then the calculated model is scaled by this value to compare it to the original model with the same scalar values. This research uses the root mean square (RMS) of the three dimensional displacement from the original points to describe the quality of the model:

$$RMS = \sqrt{\frac{\sum_{k=1}^{n} \delta x_k^{\,2} + \delta y_k^{\,2} + \delta z_k^{\,2}}{n}}, \tag{4-2}$$

where n is the number of point correspondences and δx, δy, δz are the differences in the x, y, and z coordinates between the true and reconstructed model. To determine the values that characterize a high quality model, the root mean square values from one hundred iterations of models are used to calculate an average:

$$RMS_{avg} = \sqrt{\frac{\sum_{k=1}^{n} RMS_k^{\,2}}{n}} \tag{4-3}$$

4.3 Simulation Results

The first part of this section discusses findings when perfect correspondences were used; the second part discusses how well the algorithm handles noise.

4.3.1 Perfect Correspondence

The algorithm was run on the different sets of perfect correspondences from the virtual images described in Chapter III. Figure 4-2 shows results for full Euclidian reconstruction from the initial image and the image where the camera is translated along

the x, y, and z axes by one unit. The Euclidian structure is apparent in the model and confirmed by an RMS_{avg} that equals zero.

There are some cases where a Euclidian structure is not attained. These cases are shown in Appendix C and are the result of an unbounded image plane. In these tests, every point appears in every image, but some points should have been rotated out of the image; they are behind the camera due to the rotation. When features that are rotated into the negative image still appear in the image, the algorithm breaks down and produces models like the one shown in Figure 4-3. This algorithmic failure is a result of the synthetic images and should not appear when images taken from a photographic device are used.

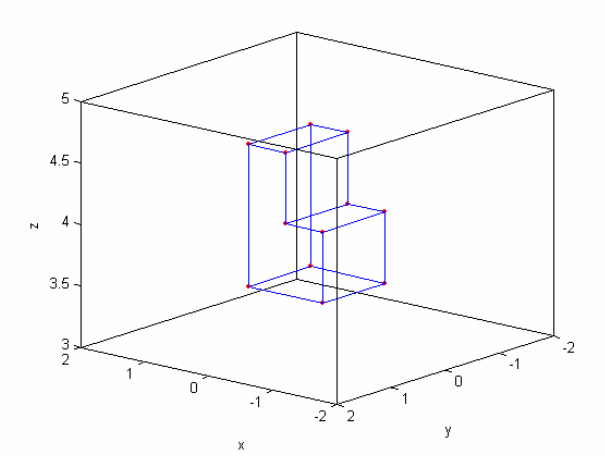


Figure 4-2 – Euclidian Reconstruction with Zero Noise Level

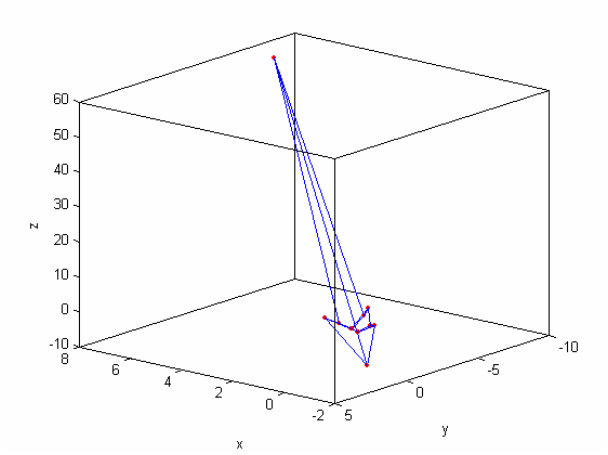


Figure 4-3 – Euclidian Reconstruction with Zero Noise Level and Weak Motion

Results of tests from the first set of images reveal two findings. First, the structure from motion algorithm can recreate the simple model from sets of correspondences, up to a scalar factor. They also revealed that the algorithm is not capable of recreating a Euclidian structure if the points are not in front of the camera.

Results from the second set of images show how sweep angle affects the model produced by the structure from motion pipeline. The sweep angle is the angular value between the two vectors from the different camera locations to a feature (see Figure 4-4). Figure 4-5 shows the progression of a model that was created from the simulation

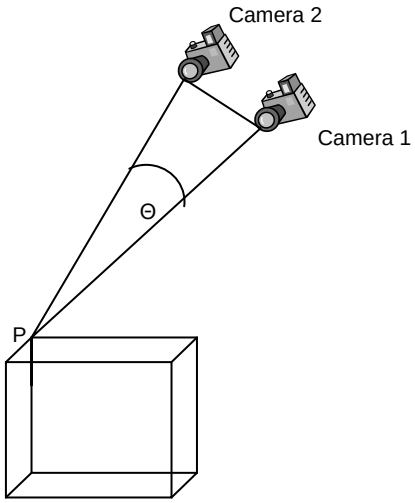
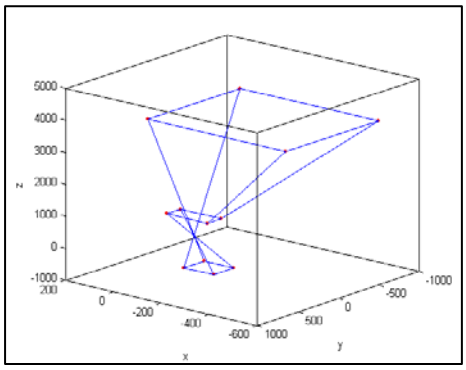
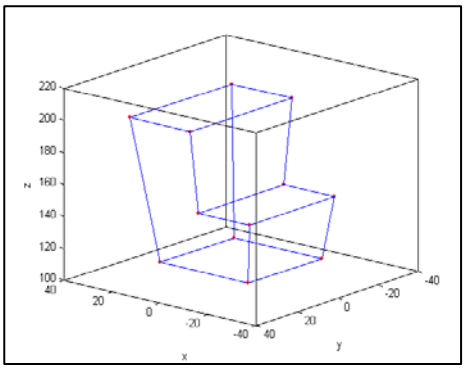


Figure 4-4 – The Sweep Angle (θ) between two Camera Locations

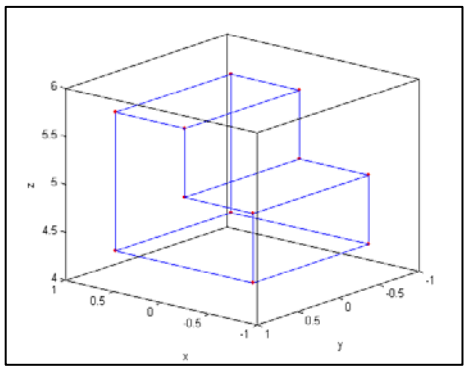
data as this angle is increased. For camera motions that only involve the *z* axis this value



Model with 6° between cameras



Model with 10° between cameras



Model with 15° between cameras

Figure 4-5 – Models as the Sweep Angle Between Images Increases

is always zero and produces a distorted model. To better characterize the effect of the sweep angle, the RMS_{avg} is calculated for every reconstruction done with the images from set two, and the results are shown in Figure 4-6. For these cases a noise standard deviation of 1 pixel is used. The high RMS_{avg} values shown at low sweep angles present a problem when using aerial imagery. Since aircraft fly at high altitudes, subsequent images taken result in low sweep angles. To compensate for this, additional tests were done, substituting real translations and rotations of the camera into the Euclidian

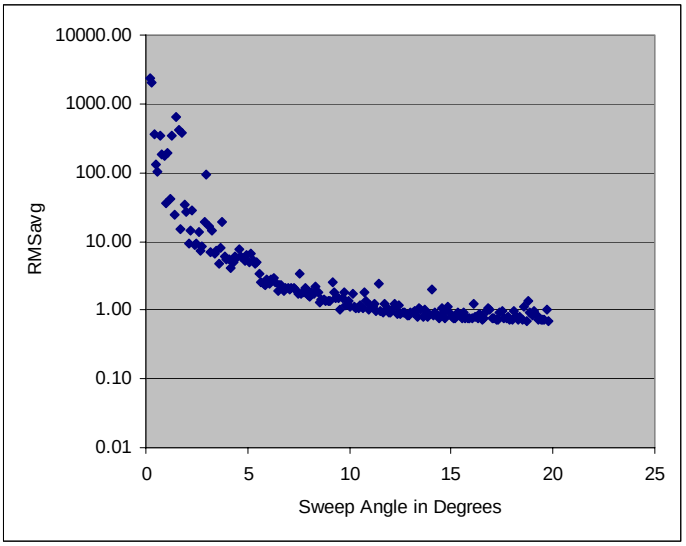


Figure 4-6 – The Sweep Angle (θ) compared to RMS_{avg} in Full Euclidian Reconstruction

reconstruction process instead of the values estimated from the correspondences. Figure 4-7 shows the results of these tests. Even at low sweep angles, when true rotation and translation is used instead of estimated rotation and translation, the reconstruction produces accurate results.

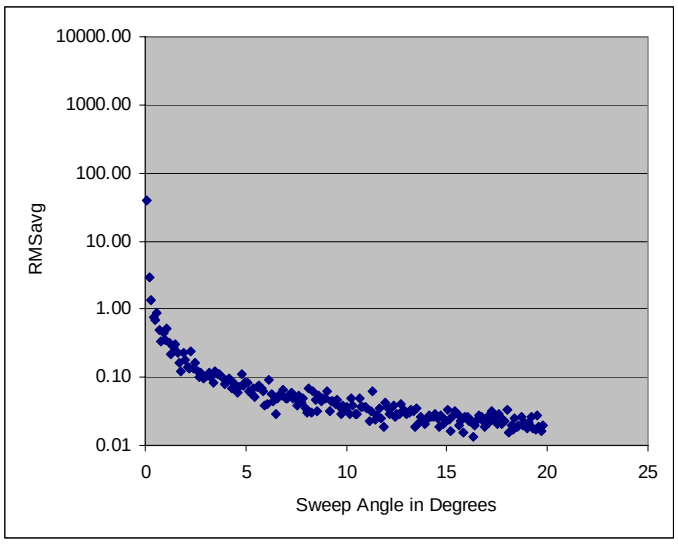


Figure 4-7 – The Sweep Angle (θ) compared to RMS_{avg} in Navigation Aided Euclidian Reconstruction

A comparison of Figure 4-6 and Figure 4-7 shows the extreme value of externally calculated camera motion derived from the navigation information of an aircraft. This value pertains to a three dimensional reconstruction from a structure from motion pipeline using aerial images (which tend to have small sweep angles). At low sweep angles, the RMS_{avg} for a full reconstruction is two orders of magnitude larger than the corresponding navigation aided reconstruction.

These results define two properties for camera movement that result in a Euclidian model when structure from motion is accomplished. The first property is that the camera must point toward the points of interest. The second property is that the sweep angle between the vectors from the camera to the features must be large enough to provide the information necessary to estimate the motion undergone by the camera (see Figure 4-4 and Figure 4-6). It is also discovered that the sweep angle deficiency can be

overcome by including the real translation and rotation that the camera underwent between the two images.

These findings from the perfect correspondence test are particularly important for examining how the algorithm handles aerial imagery. Users of the proposed implementation of an aerial imagery structure from motion pipeline must emphasize the importance of rich motion in the movement of the aircraft to flight planners. If the motion of the aircraft and the camera does not provide enough information, the algorithm produces a distorted model. If this is the case, then the real rotation and translation of the camera must be used to produce an accurate model.

4.3.2 Noise Introduction

Noise was added to the correspondences for the image created from the translation described in Case 4 in Table 4-2. Five different levels of noise were used. Each level corresponds to the standard deviation for generating the random pixel displacement value from the Gaussian distribution:

- Noise level 1 used $\sigma = 1$
- Noise level 5 used $\sigma = 5$
- Noise level 10 used $\sigma = 10$
- Noise level 15 used $\sigma = 15$
- Noise level 20 used $\sigma = 20$

At each level, 100 reconstructions were accomplished. Then the *RMS* (from Eq. 4-2) was calculated for each reconstruction. These values were used to calculate the RMS_{avg} (from

Eq. 4-3) for each noise level. The RMS_{avg} value for each level shows the effect of that amount of noise on the model. Figure 4-8 illustrates the exponential trend of the effect.

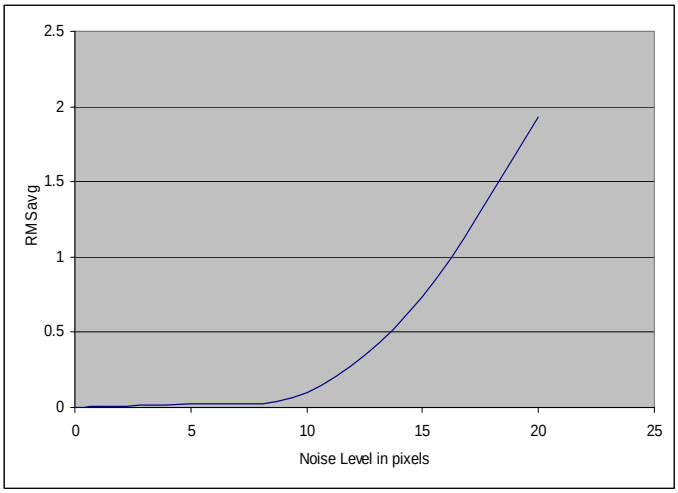


Figure 4-8 – Error as Noise is added to Pixel Correspondences from the Simulation Testing

This data shows that the method used to calculate the three dimensional structure is capable of handling small amounts of noise and of still producing a model that is similar in structure to the actual model (see Figure 4-9). The level of acceptable noise has a direct impact on the type of feature correspondence algorithm used in the feature tracking section of the pipeline.

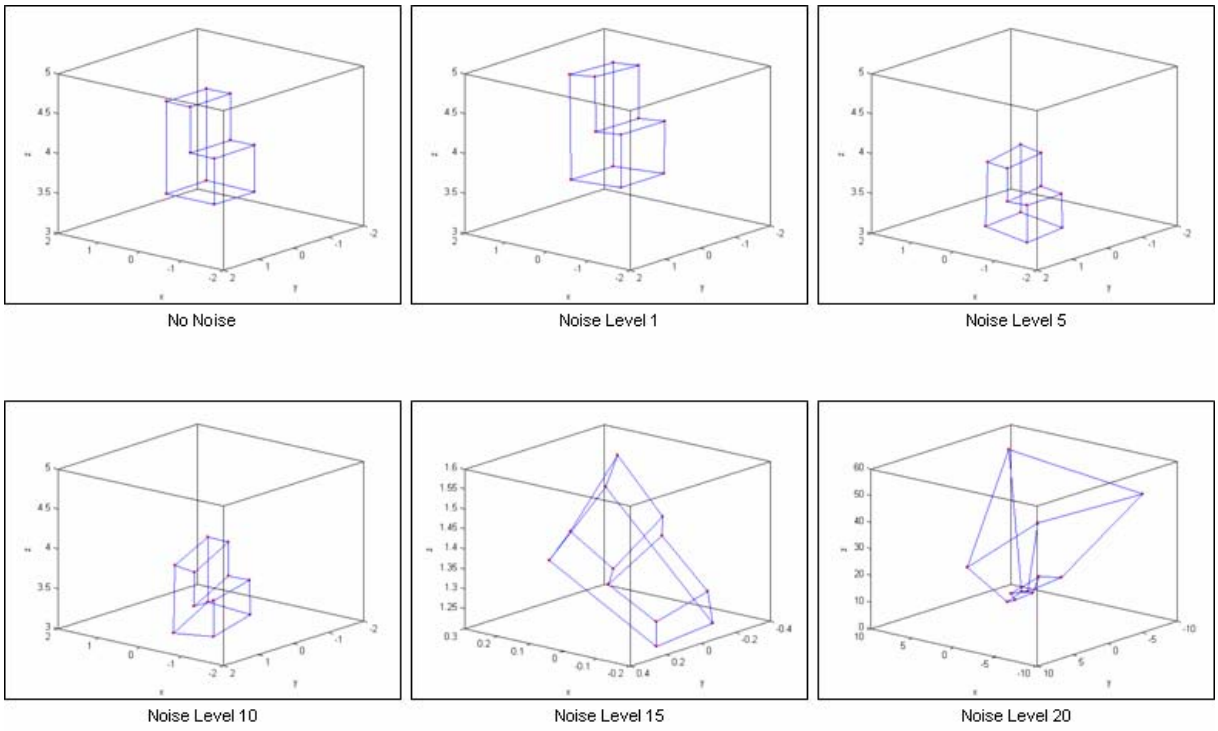


Figure 4-9 – Structure Created from Different Noise Levels

4.4 Flight Test

The flight test used aerial imagery. The sponsor of this research, the Air Force Research Laboratory/Munitions Group (AFRL/MNG), provided video from one of their experiments at the Marine Urban Warfare Center. This video was taken from a blimp as it flew around the compound. The blimp had a GPS receiver, a commercial digital video camcorder with analog GPS timestamp overlay, an inertial navigation system, and a laser-detection and ranging (LADAR) camera on board to create more sophisticated three-dimensional imagery of the center. These capture devices provided the location of the camera along with the video inputs for testing and verifying the implemented algorithm. During the taping of the video, the on site crew took GPS measurements from

various points of interest around the compound, ensuring that they appear in the blimp flight path.

Using the camera calibration method described in Chapter III and these points of interest, the intrinsic camera parameters was estimated. The digital video and estimated camera matrix *K* was then used as inputs for the structure from motion algorithm.

To verify the accuracy of the model, the vectors from the camera to the points of interest were calculated using the GPS data. These vectors were then compared to the results from the algorithm.

4.5 Results of Camera Calibration

The camera calibration algorithm was implemented as described in Chapter III. The algorithm converges when any possible change to the parameters increases the distance from calculated pixel coordinates to the actual pixel coordinates. This convergence yielded the intrinsic parameters for the camera that produced the images. The parameters make up the following *K* matrix and were used in the flight tests to calculate the three-dimensional structure from the video provided by AFRL/MNG:

$$K = \begin{bmatrix} 2782 & 0 & 0 \\ 0 & 2447 & 480 \\ 0 & 0 & 1 \end{bmatrix}. \tag{4-4}$$

To ensure accuracy of the results, the calculated pixel coordinates shown in Figure 4-10 were compared to the actual pixel coordinates shown in Figure 4-11.

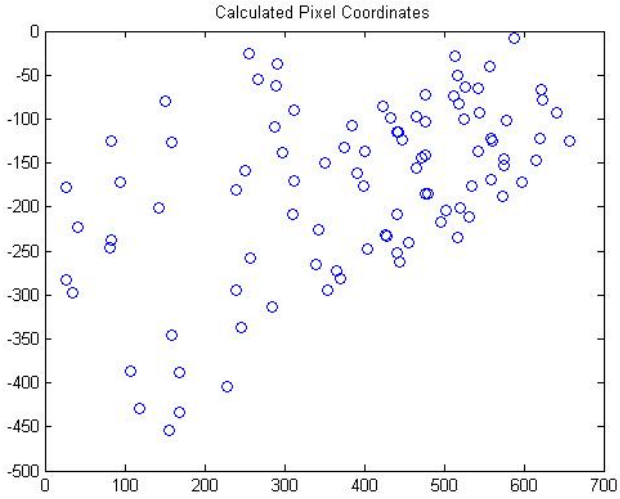


Figure 4-10 – Calculated Pixel Coordinates

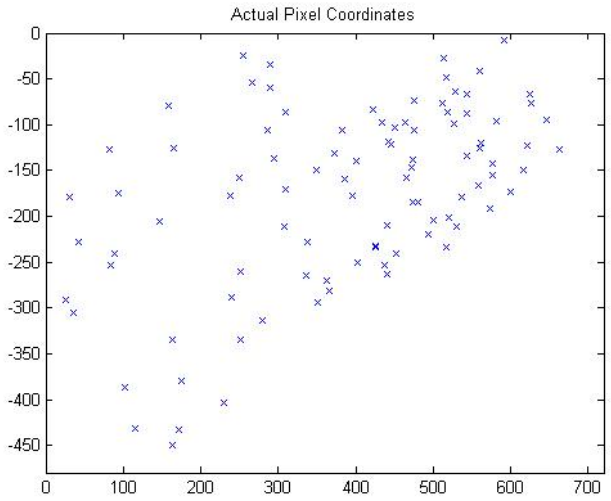


Figure 4-11 – Actual Pixel Coordinates

The accuracy of the algorithm is apparent when the actual pixel coordinates are overlaid with the calculated pixel coordinates (see Figure 4-12). Recall that these points are generated from five different images at different positions and orientations. The small number of calculated pixel coordinates that do not match with the actual pixel

coordinates and the distance that they are displaced show the high accuracy of the estimated camera parameter matrix.

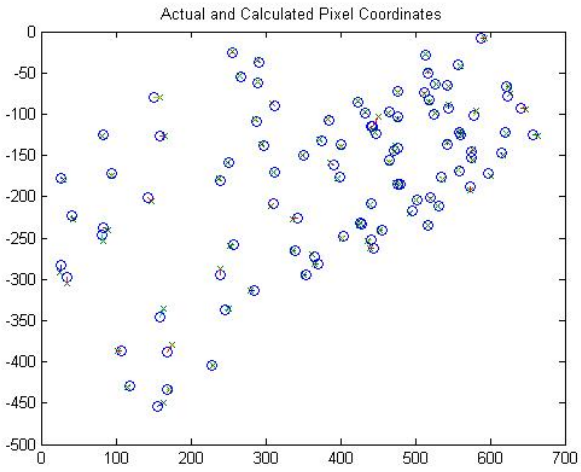
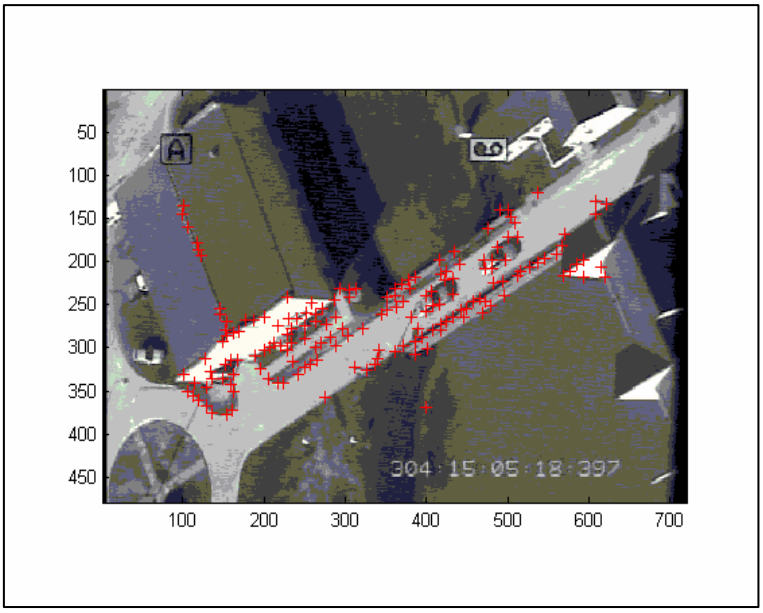


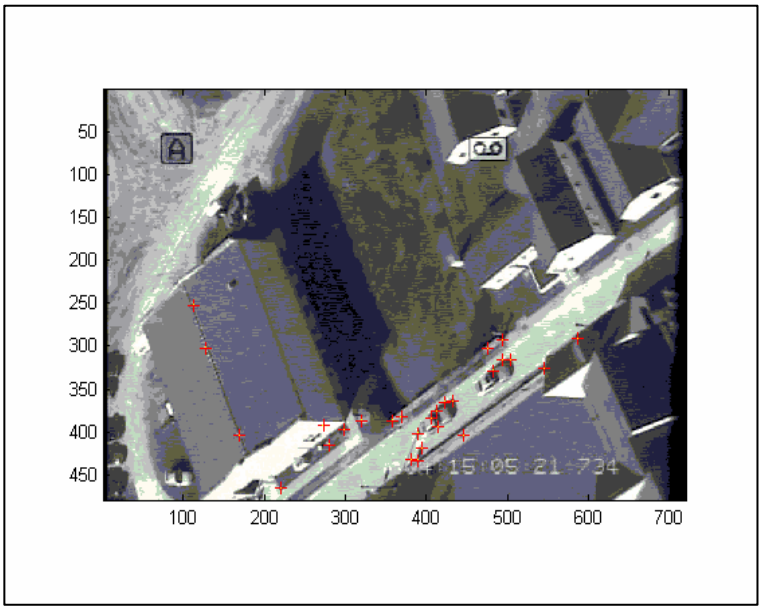
Figure 4-12 – Combined Actual and Calculated Pixel Coordinates

4.6 Flight Test Results

The structure from motion pipeline was tested with aerial imagery to verify that the pipeline is capable of producing acceptable models. The video provided by AFRL was used as input to the pipeline along with the results from the camera calibration algorithm. The top portion of Figure 4-13 shows the first image with the features the selection algorithm used for tracking. The bottom portion of Figure 4-13 shows the last image of the sequence with the features successfully tracked by the algorithm.



Start Image with 167 Selected Features



End Image with 26 Features Successfully Tracked

Figure 4-13 – Images used for Flight Tests

Figure 4-14 shows the resulting model. It is easy to see extreme distortion in the model, which results from the geometry of aircraft motion. The distance of the camera in image

one to the camera in image two is less than twenty meters, making the angle between

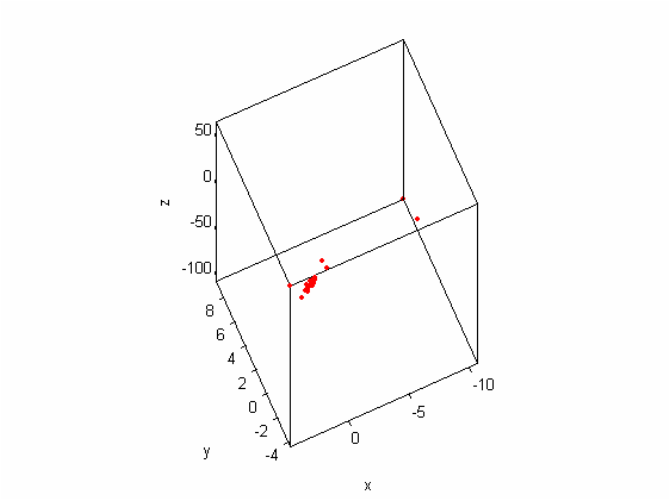


Figure 4-14 – Model from Structure from Motion Pipeline using Full Euclidian Reconstruction

them and any point in the image less than 4° (The data from the simulation tests verifies this result). Figure 4-5 and Figure 4-6 show that a sweep angle of 4° is too small to provide enough motion data to reconstruct camera movement.

After reviewing all of the video provided by AFRL/MNG it was determined that there are no acceptable image pairs for performing the complete pipeline. An acceptable image pair is a set of images that contains enough (eight or more) feature correspondences and that has enough movement between the camera locations (i.e., a large enough sweep angle) to estimate camera motion.

To compensate for the lack of acceptable image pairs, further tests were performed with the known point correspondences from the camera calibration algorithm. To accomplish these extra tests, the rotation and translation from image one to image three from Table 3-1 was calculated from the GPS and INS information collected during

the taping. These motions were then substituted instead of the estimated rotation and translations into the Euclidian portion of the algorithm using the known correspondences. This substitution was based on the navigation-augmented Euclidian reconstruction tests outcome (see Figure 4-7). Figure 4-15 shows the resulting model in the world coordinates frame marked with ‘o’s and solid lines, and the true points marked with ‘x’s and dashed lines. These points correspond to the features marked in Figure 3-2.

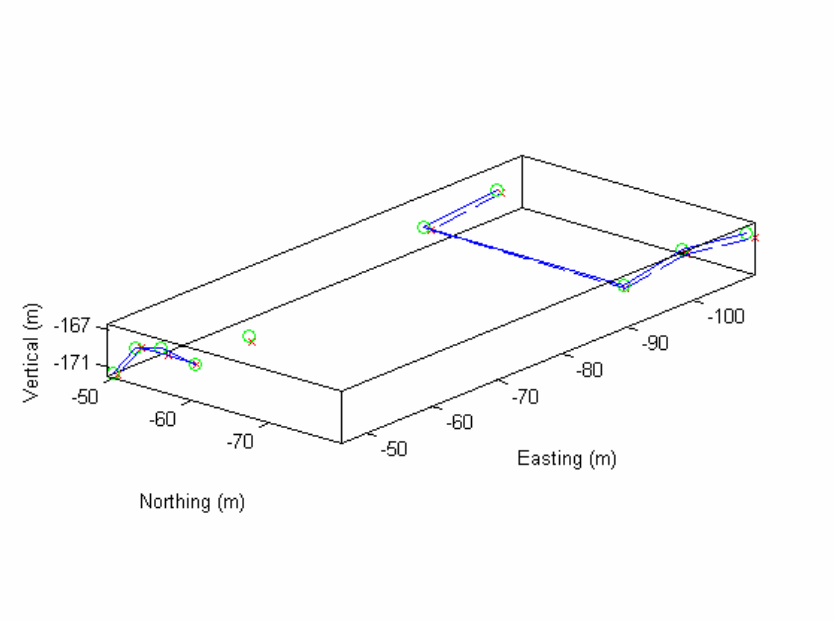


Figure 4-15 – Calculated and Actual Points from Navigation-Aided Euclidian Reconstruction on Aerial Imagery

The root mean square of the displacement in three-dimensional space of the points is 0.66m. This small level of displacement can be attributed to human errors made while performing the point correspondences. This error becomes apparent when results are evaluated according to the image plane projections. Table 4-3 shows the calculated pixel values features with the estimated pixel values that

Table 4-3 – Comparison of Calculated pixel Values to Human Detected Pixel Values

	Human Detected		Calculated	
Point	X	Y	X	Y
bc2002	164	335	161.02	342.13
bc2003	240	288	241.6	291.48
bc2004	159	79	152.36	76.39
bc2005	82	127	85.228	121.77
bc2006	30	179	29.563	174.35
bc2301	473	138	478.21	137.76
bc2308	627	76	625.9	74.776
bc2309	581	96	580.95	97.922
bc2310	561	120	563.15	121.13
bc2311	544	88	546.32	88.49

are selected from one of the images. These results indicate that structure from motion can be accomplished on aerial imagery and answer the first investigative question.

4.7 Summary

This chapter describes the simulation tests and the flight tests performed to evaluate the implemented structure from motion pipeline. The test sets showed that this research's implementation of the structure from motion pipeline reported here works for aerial imagery. The tests also characterized some motions that are incapable of recovering three-dimensional structure.

V. Conclusions and Recommendations

5.1 Conclusions of Research

This research examines the structure from motion pipeline. It does so to determine whether or not the pipeline is applicable to images captured from airborne cameras. Once the aerial capability of the structure from motion pipeline is discovered, the limitations of the pipeline when combined with the unique movements associated with flight are investigated.

The investigation consists of two steps, simulation tests and flight tests. The simulation tests are accomplished using the three-dimensional model. Then synthetic images are taken of the three-dimensional model using the camera model described in Chapter II and Chapter III. These image sets have three purposes in this research. First, they show that the algorithm is correctly implemented. Second, they describe motions that are rich enough to produce Euclidian models. The first set of synthetic images show the limitations on the rotation of the camera which ensure that enough features stay within the image. The second set of synthetic images characterizes the minimum sweep angle necessary for a Euclidean reconstruction. This image set shows that small sweep angles produce warped models of the three-dimensional structure. It also illustrates the benefits of using the navigation-aided Euclidian reconstruction. At small sweep angles the RMS_{avg} is two orders of magnitude larger when using the full Euclidian reconstruction methods. Finally, both image sets provide a framework to develop the RMS_{avg} comparison model used to check the accuracy of the flight testing step.

The flight test step includes running the pipeline on real aerial imagery. Due to the geometry of blimp movement, the angle between the images is too small to estimate the motion of the camera from the feature correspondences. This results in drastically warped models. However, to establish the validity of an aerial-imagery based structure from motion system, further testing is accomplished. The actual movement of the blimp is recovered using the INS and GPS data. Those values are substituted, instead of the estimated motion parameters, into the triangulation section of the Euclidian structure recovery. The resulting model reflects the true Euclidian structure of the features selected. The average three-dimensional displacement of the points is 0.66m when using the navigation-aided methods. This result demonstrates the benefit of using actual navigation measurements when reconstructing models from images.

5.2 Recommendations for Future Action

This research establishes that structure from motion can be accomplished on aerial imagery. There are some limitations to the process—the intrinsic camera parameters must be known and the motion of the camera must be rich enough to determine the rotation matrix and translation vector. Further research should address one or more of the following goals

- Accomplishing more tests with a larger variety of imagery to better characterize camera movements that are rich enough to determine the camera motion
- Establishing a system that can automatically perform camera calibration and incorporating this process into the current system

- Developing stronger feature detection and tracking algorithms to increase the speed of the system.
- Accomplishing more tests using the navigation augmented approach rather than the straight structure from motion pipeline.

Appendix A – AFRL/MNG Documentation

The following is the readme file provided with the aerial imager by AFRL/MNG.

Readme File for AFRL/MNG Video-Enabled Autonomous Agents (VEAA) Data Set #1

This data set contains aerial video, measured 3D point clouds, and GPS/INS data appropriate for computer vision research in areas such as optical flow, passive 3D, and structure from motion.

Video Camera Information:
Video imagery was collected using a commercial DV camcorder with analog GPS timestamp overlay. The camera was side mounted ~90 degrees to the horizon and at a ~45 degree depression angle. A GPS/INS system was collocated with the camera to provide motion truth. Camera imagery is 720x480 resolution with an approximate FOV of 7.33 x 5.5 degrees. The DV imagery is interlaced and compressed using the Microsoft MPEG-4 V2 codec.

3D Point Cloud Information:
3D measured point clouds were captured for some video sequences. The point files are ASCII and ordered in columns of Easting, Northing, Height Above Ellipsoid, Intensity. Point files are denoted by a ".xyz" suffix.

GPS/INS Information:
A commercial differential GPS/INS was used to provide motion truth. The IMU was collocated with the camera, while the GPS antenna had some separation. The camera is roughly 0.5m behind, 1.5m left, and 0.75m below the GPS antenna and has a ~2 degree error in boresight in the clockwise yaw direction. GPS/INS data is provided for each video sequence in a Matlab .mat file. The data is stored in a self-documenting structure. Units are in UTM. The GPS/INS data has been time registered to the video such that each entry in the GPS/INS data directly corresponds to video frame.

Ground Truth File:
"Ground Truth.xyz" provides a sparse array of building corner and fiducial locations in an ASCII column format. Column 1 provides a point label followed by Easting, Northing, and Height Above Ellipsoid. These points were surveyed using a differential GPS.

Inquires concerning this data should be directed to -
AFRL/MNGI Computational Vision Team
101 Eglin Blvd Suite 205
Eglin AFB, FL 32544

Or via e-mail to mailto:XXX@eglin.af.milzetterli@eglin.af.mil

Use of this data should be cited as AFRL/MNG VEAA Data Set #1 in publications and presentations.

Appendix B – Image Timestamp and Location Data for Camera Calibration Algorithm

Point Identifier	Easting from UTM 18 (m)	Northing from UTM 18 (m)	Altitude (HAE) (m)
bc2001	293851.68	3838192.51	-18.58
bc2002	293851.87	3838201.55	-17.53
bc2003	293852.04	3838212.02	-18.53
bc2004	293826.63	3838212.52	-18.55
bc2005	293826.37	3838203.46	-17.51
bc2006	293826.22	3838193.05	-18.59
bc2101	293855.65	3838258.63	-22.77
bc2102	293855.75	3838262.44	-21.11
bc2103	293855.75	3838266.13	-22.96
bc2104	293852.14	3838266.30	-22.97
bc2105	293852.19	3838274.78	-23.01
bc2107	293845.27	3838274.89	-22.99
bc2108	293845.10	3838266.38	-23.05
bc2108	293845.10	3838266.38	-23.05
bc2109	293843.29	3838266.32	-22.95
bc2110	293843.24	3838262.58	-21.11
bc2111	293843.20	3838258.85	-22.98
bc2301	293851.74	3838239.82	-22.42
bc2302	293848.84	3838239.59	-19.93
bc2303	293844.75	3838239.91	-22.40
bc2304	293844.87	3838248.41	-22.44
bc2305	293843.10	3838248.44	-22.41
bc2306	293843.08	3838252.28	-20.49
bc2307	293843.13	3838255.97	-22.38
bc2308	293855.55	3838255.86	-22.40
bc2309	293855.66	3838252.04	-20.58
bc2310	293855.51	3838248.28	-22.43
bc2311	293851.88	3838248.26	-22.45
bc2701	293834.88	3838255.33	-22.40
bc2702	293834.35	3838251.73	-22.44
bc2703	293837.88	3838251.15	-22.41
bc2704	293837.24	3838246.81	-20.22
bc2705	293836.54	3838242.48	-22.38
bc2708	293824.40	3838253.26	-22.38
bc2709	293828.02	3838252.74	-22.41
bc2710	293828.59	3838256.28	-22.39

Appendix C – Rotations from Perfect Correspondences Simulation Test that Rotate Correspondences out of the Image Plane

Test Number	T(x)	T(y)	T(z)	R(x)	R(y)
9	1	1	1	0	60
10	0	1	1	0	60
11	1	0	1	0	60
12	1	1	0	0	60
13	1	1	1	0	90
14	0	1	1	0	90
15	1	0	1	0	90
16	1	1	0	0	90
25	1	1	1	30	60
26	0	1	1	30	60
27	1	0	1	30	60
28	1	1	0	30	60
29	1	1	1	30	90
30	0	1	1	30	90
31	1	0	1	30	90
32	1	1	0	30	90
33	1	1	1	60	0
34	0	1	1	60	0
35	1	0	1	60	0
36	1	1	0	60	0
37	1	1	1	60	30
38	0	1	1	60	30
39	1	0	1	60	30
40	1	1	0	60	30
49	1	1	1	90	0
50	0	1	1	90	0
51	1	0	1	90	0

Bibliography

[1] Baumberg, Adam. “Reliable Feature Matching Across Widely Separated Views.” In *Conference on Computer Vision and Pattern Recognition*. pages 1774-1781. 2000.

[2] Habed, Adlane and Boubakeur Boufama. “Three-dimensional Projective Reconstruction from Three views.” In *International Conference on Pattern Recognition (ICPR'00)-Volume 1*. pages 1415-1418. 2000.

[3] Harris, C. and M. Stephens. “A Combined Corner and Edge Detector.” In *Proceedings of the Alvey Conference*. pages 189-192. 1988.

[4] Heijden, Ferdinand van der. “Edge and Line Feature Extraction Based on Covariance Models.” In *IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol 17 , No 1*. pages 16-33. 1995.

[5] A. Heyden and K. Astrom. “Euclidean reconstruction from image sequences with varying and unknown focal length and principal point.” In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 438-443, 1997.

[6] Ishikawa, Hiroshi. “Multi-scale Feature Selection in Stereo.” In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. pages 132-137. 1999

[7] Kermad, Chafik and Christope Collewet “Improving Feature Tracking by Robust Points of Interest Selection.” In *6th International Fall Workshop on Vision, Modeling and Visualization*. 2001.

[8] Ma, Yi, Stefano Soatto, Jana Kosecka, S. Shankay Sastry. An Invitation to 3-D Vision: From Images to Geometric Models. Springer. New York : 2004.

[9] Meer, Peter. “Edge Detection with Embedded Confidence.” In *IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol 23 , No 12.* pages 1351-1365. 2001.

[10] Raquet, John and Michael Giebner. “Navigation Using Optical Measurements of Objects at Unknown Location.” In *ION 59th Annual Meeting/CIGTF 22nd Guidance*. Pages 282-290. 2003.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 074-0188

1. REPORT DATE *(DD-MM-YYYY)*	**2. REPORT TYPE**	**3. DATES COVERED** *(From – To)*
21-03-2005	Master's Thesis	March 200 – March 2003

4. TITLE AND SUBTITLE

Determination of Structure from Motion Using Aerial Imagery

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Graham, Paul R., First Lieutenant, USAF

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAMES(S) AND ADDRESS(S)

Air Force Institute of Technology
Graduate School of Engineering and Management (AFIT/EN)
2950 Hobson Way, Building 640
WPAFB OH 45433-8865

8. PERFORMING ORGANIZATION REPORT NUMBER

AFIT/GCS/ENG/05-06

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Virgil E Zetterlind, Capt
AFRL Munitions Directorate, Advanced Guidance Division,
Seeker Image and Signal Processing Branch
101 Eglin Blvd Suite 210
Eglin AFB, FL 32542
virgil.zetterlind@eglin.af.mil

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

The structure from motion process creates three-dimensional models from a sequence of images. Until recently, most research in this field has been restricted to land-based imagery. This research examines the current methods of land-based structure from motion and evaluates their performance for aerial imagery.

Current structure from motion algorithms search the initial image for features to track though the subsequent images. These features are used to create point correspondences between the two images. The correspondences are used to estimate the motion of the camera and then the three-dimensional structure of the scene. This research tests current algorithms using synthetic data for correctness and to characterize the motions necessary to produce accurate models. Two approaches are investigated: full Euclidian reconstructions, where the camera motion is estimated using the correspondences, and navigation-aided Euclidian reconstructions, where the camera motion is calculated using the Global Positioning System and inertial navigation system data from the aircraft.

Both sets algorithms are applied to images collected from an airborne blimp. It is found that full Euclidian reconstructions have two orders of magnitude more error than navigation-aided Euclidian reconstructions when using typical images from airborne cameras.

15. SUBJECT TERMS

Structure from Motion, Computer Vision, Aerial Imagery

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON**
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**			John F. Raquet, Ph.D.
U	**U**	**U**	**UU**	61	**19b. TELEPHONE NUMBER** *(Include area code)* (937) 255-3636, ext 4580 (john.raquet@afit.edu)

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