A LATERAL VISION-BASED CONTROL AUTOPILOT FOR MICRO AIR VEHICLES USING A HORIZON DETECTION APPROACH

By

RYAN SCOTT CAUSEY

A THESIS PRESENTED TO THE GRADUATE SCHOOL
OF THE UNIVERSITY OF FLORIDA IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF SCIENCE

UNIVERSITY OF FLORIDA

2003

I dedicate this work to my lovely wife, Liza P. Causey, that has supported me every step of the way. Her love and understanding through the years have brought my passion for life beyond boundaries.

ACKNOWLEDGMENTS

I want to acknowledge and thank AFRL-MN and Eglin Air Force for the funding of this project. I want to thank all the research members involved with this project for their dedication and long hours in the lab and in the field which includes Dr. Peter Ifju and his team Kyuho Lee, and Sewoong Jung that developed and fabricated the MAVs, Dr. Michael Nechyba and his team Jason Grzywna, Jason Plew, and Ashish Jain that did an extraordinary job designing and developing all the on-board and ground station hardware along with the horizon detection algorithm, my colleagues in the Dynamics and Controls Lab: Helen Garica, Kristin Fitzpatrick for her *S* flight, the new graduate students Joe Kehoe and Jason Jackowski. A special thanks to the pilots Kyuho Lee and especially Mujahid Abdulrahim for his advice and extremely long flight hours, and most importantly Dr. Andrew Kurdila and Dr. Rick Lind for their remarkable guidance and inspiration that will truly last a life time and finally to my parents James and Sandra Causey who provided me the guidance and discipline needed for success.

TABLE OF CONTENTS

LIST OF TABLES

Table page

LIST OF FIGURES

Figure page

Abstract of Thesis Presented to the Graduate School
of the University of Florida in Partial Fulfillment of the
Requirements for the Degree of Master of Science

A LATERAL VISION-BASED CONTROL AUTOPILOT FOR MICRO AIR VEHICLES USING A HORIZON DETECTION APPROACH

By

Ryan Scott Causey

December 2003

Chair: Andrew J. Kurdila
Cochair: Richard C. Lind, Jr.
Major Department: Mechanical and Aerospace Engineering

The concept of autonomous flight has been realized as a feasible concept with the current technology, where small electronics have advanced to standard equipment that is inexpensive. The aircraft platform to head these efforts has shifted to small agile vehicles that can maneuver within confined environments such as urban terrains. Some capabilities needed for autonmony include sensors that detect the physical surroundings, data processing, path planning, and the control algorithms. Micro air vehicles (MAVs), structurally have evolved to house the required payload for autonomous research. The capabilities of MAVs have demonstrated maneuverability for confined spaces, on-board processing, and camera integration with vision feedback. This thesis will establish the required analytical relationships for a camera on-board an MAV, along with image processing for feature extraction, and finally control design and implementation using this physical feature. The relationship first derived, couples the nonlinear equations of motion of an aircraft to the trajectory of feature points in the image plane for cameras attached to the vehicle. Algorithms using statistical models for pattern recognition are used to define the horizon in the image plane. Based on

geometry, this horizon line is used to calculate a roll estimate and pitch percentage which are used as feedback to design controllers. A testbed was fabricated along with a particular MAV and a single camera of known parameters to implement vision-based lateral control. The camera nonlinearities were studied for this particular feature (i.e. the horizon) through some observations. By using the horizon, these nonlinearities were able to be neglected from the controls system. Numerous flight tests were then conducted to improve the lateral control design by adjusting gains to acquire a desired response.

CHAPTER 1
INTRODUCTION

1.1 Motivation

Micro air vehicles, or better known as MAVs, are currently the platform for research in autonomous flight at the University of Florida. A typical MAV scale consists of a wingspan and fuselage that range from 30 inches down to 6 inches and operate in flight speeds of 25 *mph* or less [31]. The motivation behind Micro Air Vehicles are their mission capabilities in dangerous environments. For instance, MAVs are being considered for missions that involve surveillance, damage assessment, search and rescue, and region mapping. The interest behind using MAVs are their fundamental features. These vehicles are highly agile and can bank and climb within the confined space of an urban environment. Because of this feature, MAVs have the ability to change the environment in which standard aircrafts have been known to operate. For instance, MAVs can be used fly within and around the confines of buildings and subways. Additional benefits of MAVs, regarding implementation, are readiness to fly and a compact size for storage. Most importantly, MAVs are low cost, which would allow multiple MAVs to be standard equipment for a variety of users, ranging from soldiers on the battle field to police and fire agencies in our local districts.

An autonomous MAV is particularly suited to accomplish missions with minimal risk to personnel. Consider a scenario of particular concern where a biological agent has been released in an urban environment. Current operations require humans gather equipment and venture into hazardous regions. Instead, future operations could be conducted by autonomous MAVs deployed from a nearby emergency station. MAVs would assist in a diverse range of missions including detection, mapping, and search

and rescue during this type of situation. In the case of detection, the vehicle could transport a sensor that detects infectious agents. A map of the region and the infected areas could then be generated to inform civilians from real time updates. The images could also be used as a surveillance tool to search and locate possible survivors. The vehicle's small size and agility would allow access inside buildings and subways therefore increasing the search area. Analyzing image information, rescue teams can assess damage to the surroundings to plan a safe rescue route. This scenario, along with many others, validates the growing interest for such sophisticated systems using vision.

1.2 Background

1.2.1 Micro Air Vehicles

Over the past five years, MAV designs have been developed to improve flight characteristics, payload capacity, and structural integrity. With these significant improvements, very small vehicles are being considered an achievable technology for autonomous flight with many applications. Here, autonomy is regarded as the controls system making decision on the trajectory without human intervention.

The size constraints involved in using such small vehicles has lead to many advances in miniaturized digital electronics, communications, and computer technologies to make autonomous MAVs a reality [46]. These developments have allowed MAVs to be equipped with the latest computer and sensor technology such as video cameras, data processors, and communications devices. Over the last few years, the integration of these electronics onto an MAV has made considerable advances toward the overall goal of on-board processing and mission capability in real time.

The first successful MAV that demonstrated mission capabilities was Aerovironment's *Black Widow* [13]. This 6 *in* MAV was electrically powered and consisted of a rigid-wing design with three vertical stabilizers. The on-board systems included a data logger sampling 16 channels at 20 Hz and a custom video camera system that

optimized weight, power and size with image quality. A yaw damper was implemented to damp out a low frequency Dutch roll oscillation to improve not only flight stability but also stabilizing the video to obtain a steady image. An autopilot system was also incorporated using three modes: dynamic pressure hold, altitude hold, and heading hold. The final overall system, with a mass of 80 *g*, was able to sustain 30 *min* flights for a maximum range of 1.8 *km* while transmitting video images to the ground. These images were used by the pilot to control the aircraft when the range of the vehicle became too large from an RC pilot's standpoint. Most importantly, the images demonstrated the functional capability of MAVs by streaming real time video to the ground to establish their role in implementing vision-based autonomous flight.

The rigid-wing concept was common in many other successful designs including the *Trochoid* developed by MLB Company and *Microstar*, also funded by DARPA [17]. The disadvantage to using rigid-winged MAVs is they require additional gyro stabilizers to assist in controlling the aircraft. This is due to the unsteady aerodynamic loading that dominates a vehicle at such low Reynolds numbers. For example, wind gust will have a much greater affect on MAVs than larger aircrafts because of their size and speeds at which they operate. Research eventually lead to the concept of flexible wings that are able to adapt to unsteady flight conditions.

Dr. Peter Ifju and his team, at the University of Florida, has developed a wing concept that consists of an extensible latex rubber membrane that is overladed onto a carbon fiber skeleton for structural support [17]. This concept was taken from sail vessels which use the sail twist to extend the wind range producing a more constant thrust even in gusty conditions [45]. An illustration of this wing structure is depicted in Figure 1–1 for a 6 *in* MAV. The important feature that one should realize is the flexibility of these wings. The flexibility can be so dramatic that the wings can be folded around the fuselage creating a more compact storage envelope. Implementing

this feature, MAVs could then be deployed from missiles moments before impact to loiter around the target space for damage assessment.

Figure 1–1: Flexible Wing Concept on a 6 *in* MAV

The aerodynamic benefits of flexible wings appear to be significant but are currently being studied in detail. By adjusting to the air flow, flexible wings produce a more favorable aerodynamic flow in such low Reynolds number environments, as documented in the research [17, 44, 46]. This is mainly caused by maintaining an attached flow and preventing flow separation. As Reynolds number decreases, the flow around the wing surface separates faster and therefore increases the drag on the aircraft. So by implementing flexible wings, MAVs appear to have better flight qualities compared to the gyro stabilized rigid winged design.

Later, a more structural analysis and flow interaction approach using nonlinear moving boundary techniques coupled with Navier-Stokes solvers to quantify the resulting flow field [22]. Results obtained in this paper consist of vorticity structure, pressure distribution, and lift and drag characteristics associated with both membrane and rigid wings concepts for comparison. Modal analysis of this interaction has been done using Proper Orthogonal Decomposition to extract most energetic modes from the resulting flow field and the wings natural response [48].

With this flexible wing concept, researchers began an approach to analyze flight techniques seen in nature. This flight concept opened the eyes of many researchers to

mimic the flight of small birds and insects into the design of an MAV. This approach seems obvious because of their common size and the bird's ability to maneuver in confined spaces, but is extremely difficult to reproduce mechanically. The common motion between small birds is high frequency flapping to produce lift as well as directional control, which is reminiscent of Hummingbirds. Many works are pursuing flapping flight techniques with flexible wings to add agility. The mechanisms and actuators required for such a task are being explored in the lab environment [32]. The difficulty for this type of MAV consist of producing an efficient flying machine with on board actuators that operate with low power consumption. As one can imagine, the complex motions at such high frequencies will require efficient, durable, and compact actuators along with sophisticated control systems. Studies have shown comparable wingtip trajectories between their vibratory flapping testbed and those exhibited by Hummingbirds in various flight modes [32]. Currently these actuators are significantly larger than a MAV's fuselage so other techniques are being explored.

Other studies in Shyy et al. [44], began to investigate different aspects of flapping flexible wings such as wing loading, models for lift, drag and power, issues with low Reynolds number flows, and recent work with flexible structures adjusting to free-stream conditions. Some preliminary results indicate improvements in lift-to-drag ratio and aerodynamic performance over the standard rigid-winged counter-partner. However, the complete understanding of flexible wings requires the full Navier-Stokes flow model to accurately represent the coupling between flapping dynamics and flight at low Reynolds numbers. The complexities involved in this approach has lead many researchers to adopt a potential flow or thin-layer flow model even though these solutions contain inadequate details of the flow at these sizes and flight regimes.

Over the last several years, Dr. Ifju has established a systematic fabrication facility for MAVs at the University of Florida. This facility enables the rapid production of MAVs in a range of sizes and the ability to customize a design for a particular mission.

The standard MAV fleet at UF range in wingspan from 6 *in* to 24 *in* and are depicted in Figure 1–2. Notice all the MAVs in this figure consists of a common design: a carbon fiber airframe with a wing that consists of a flexible membrane and a reinforced leading edge made of carbon fiber. The airframe houses specific components needed for flight testing and typically consist of accelerometers, rate gyros, a GPS sensor, data processors and loggers, servos, and camera components.

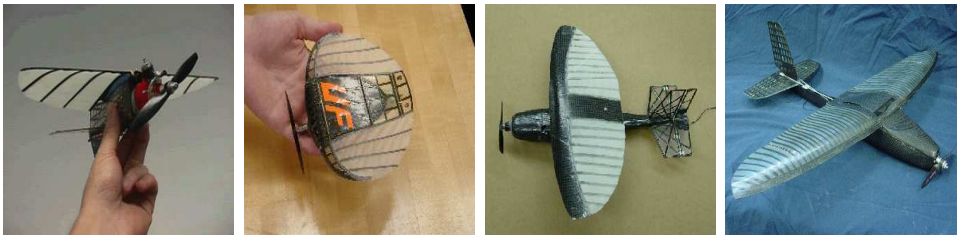


Figure 1–2: UF MAV Fleet

The control surfaces of a standard aircraft are typically incorporated on an MAV. Although, implementing an aileron on a flexible wing design has caused some difficulty during fabrication. For simplicity, UF has overcome this problem be implementing symmetric and antisymmetric elevator. This involves splitting the elevator down the center into two control surfaces. The symmetric elevator occurs when the two surfaces act together to produce pitching moments while the antisymmetric occurs when these surfaces act opposite to produce rolling moments. Recently, a new technique called morphing has evolved into the MAV design to act as ailerons. This concept will be explained in more detail in Chapter 5.1. Depending on the directional stability of each MAV design, a rudder can be implemented to reduce side slip during maneuvers and is commonly incorporated on the smaller MAVs.

To test MAVs efficiently with sophisticated control algorithms, models need to be generated from flight test or wind tunnel data. Only recently MAVs have been considered a feasible technology from a controls aspect. Researchers are now emphasizing the need for MAV models, especially nonlinear models, for control synthesis. The first MAV, similar to the one shown in Figure 1–1, was tested in a wind

tunnel, the Basic Aerodynamics Research Tunnel (BART), at NASA Langley Research Center [34]. These test provided researchers with aerodynamic coefficients and control properties to further analyze the stability. The works of Waszak et al. in [46], documented the aerodynamic performance and control properties from this wind tunnel data by investigating trim conditions and the aircraft's response to a control deflection at various angles of attack and dynamic pressures. These works proceeded by a second publication [45], in which linearized models were generated from the same data using linear regression techniques over a range of dynamic pressures. Control simulations were then constructed for these models to investigate vehicle trim and basic stability and control properties. The control structure consisted of a nonlinear dynamic inverter which converted measured angular rates to control surface deflections.

The success of the flexible wing MAV has been demonstrated, by the UF team, at the International Society of Structural and Multidisciplinary Optimization Micro Aerial Vehicle Competition by receiving the overall first place award for the last four years. This competition continues to grow every year as more universities become involved. Some other universities that are intensely pursuing MAVs include BYU, Arizona, and Notre Dame.

1.2.2 Image Processing

The task of flying autonomous MAVs in environments that are confined by buildings, trees and additional features, as well as poor weather conditions, requires a host of innovations in vision-based flight control. Two areas, in particular, that need to be addressed before vision controllers can be designed include vision processing and pattern recognition. The images acquired by a standard camera contain an enormous amount of data for processing, such as pixel color. Even MAVs traveling at speeds around 25 *mph* still require a significant number of frames per second to incorporate vision in the feedback loop. In real time, an algorithm using the complete image data for control design would be computationally expensive. Therefore, strategies in the last

several years have evolved in attempt to process and compress this data without the loss of important image features for controller design.

Image processing attempts to handle large data by extracting features from the image that are significant in the environment to reduce computation time. For example, in a city the outline of buildings would be a significant feature for an aircraft while the details of windows and doors may not be as important. The difficulty involves training smart algorithms to recognize and distinguish between important features and details that may be omitted, as well as free space and occupied space. Some techniques using fuzzy logic or statistical color matching have been studied to reduce such large data sets while determining occupied space through surface triangulation [25, 29, 30].

Once an image has been reduced within the bandwidth limits, pattern recognition is required to classify regions based on particular features using this reduced image. Some important features that may be used from an image include color and texture. Although, the common feature used for a still shot are the color components for each pixel, which are divided into red, blue and green components (i.e., the RGB color cube).

Statistical modeling of the distribution of pixel color, using Expectation Maximization algorithms (EM), is a classical technique for classification. This concept was demonstrated to detect a horizon line based on the classification of each pixel to a set of known distribution models [8]. These known distribution models are computed based on a known data set, which in this case would be a picture of the sky and ground on a particular day and is considered the training data. The models are then applied to unknown data sets (i.e., images) and decides on the classification of each pixel belonging to the sky or ground distribution. Therefore, the boundary between these two distribution is called the decision boundary, or in this case the horizon line. Kalman filters or pattern tree structures can be added to this particular case to account for small variations in color, which happens throughout the day or in different weather

conditions, and noise to add robustness. This can account for small variations but will not apply to conditions where the sky has changed color significantly, such as dusk or dawn.

Other studies in recognition have led to advances in areas such as object motion estimation and obstacle detection. Soatto, et al. have documented several methods in [37,38,39,40,41,42] which estimates the three dimensional motion of an object from a sequence of image projections commonly referred to as "Structure from Motion." These techniques pertain to nonlinear estimation and identification theory and can be solved by decoupling the states of the dynamic observer that estimates motion and the structure parameters. Obstacle detection is based on similar concepts that identifies objects traveling in and out of the image. The detection of these object and tracking requires some additional methods to estimate the motion of any pixel over time. Several recent studies [19, 27] use Sarnoff's stereo processing for obstacle detection along with Markov Random Field Models to analyze and interpret the pixel motion.

Once object detection algorithms become efficient for real-time applications, controllers can be designed using this information as feedback for obstacle avoidance in urban environments. A particular study for urban traffic situations has began to implement a fast detection algorithm to avoid hitting unseen pedestrians by using stereo vision and classification stages to control the brake system [10].

1.2.3 Vision Control

The research done to address vision-based control on vehicles has mainly been done in robotics for the 2-D case [1, 2, 15, 28]. With motion constraints in certain directions, a robot has the ability to stop it's motion before making a decision on the path based obstructions detected by the camera. This critical feature does not require the controls system to act in real-time and the vehicle is able to travel along the circumference of an obstacle. On the other hand, an airplane is required to make decisions of the environment ahead in real-time to avoid obstacles while maintaining

forward velocity required for flight. Other control examples in 2-D involve highway driving, where studies have used vision sensors to track lane dividers and obstacles within a fixed range in both forward and lateral directions [4, 6, 20, 23, 24, 26]. Some open control problems and issues with hardware development using vision feedback are assessed in Sznaier and Camps [43].

Some research has applied vision control in free floating objects such as underwater vehicles, lighter than air blimps, and space docking [7, 9, 16, 36, 35]. These studies have demonstrated object detection and tracking ability in many different objectives. A common feature in these particular vehicles is the ability to hover in a fixed position. This advantage was described above in the 2-D case where the vehicle stops it's motion to process visual data before making a decision. Helicopters are a prime example, although the controls required to hover are complex, computation time can be gained when a path is unknown. Mission capabilities for target search and tracking have been demonstrated using helicopters and are documented in the literature [3, 33]. As this vision control concept slowly progress toward aircrafts, the computations for image processing and obstacle detection for path planning in real-time become extremely difficult.

By considering vision control for aircrafts, one has to be aware of the additional degrees of freedom and performance constraints of the vehicle. The ability to turn or climb at large rates will strongly depend on the agility of the aircraft. The speed of an aircraft, even around 25 *mph*, requires fast image processing to detect and track obstacles that continuously enter the field of view. Obstacle detection for aircraft has been examined in [11] where a morphological filter was used for image processing and object detection by retaining small sized features, which were assumed targets, while removing large features, which was assumed to be the background. An estimation problem of the aircraft's motion can then be calculated using an extended Kalman filter approach to subspace constraints, which arises when points stationary

on the environment are tracked on the image plane [14]. Recent studies have gone further by integrating object detection with control designs for autonomous aerial refueling [47, 21] and state determination for night landings [5]. The simulations for these studies both documented precise target tracking for aerial refueling and the feasibility of night landings, both with good disturbance rejection.

The design of autopilots for unmanned MAVs using vision as the primary control sensor has only begun in the research community. Significant strides were made at the University of Florida when a horizon detection algorithm, described in Section 1.2.2, was used to feedback roll angle for lateral control [8]. The control design used in this paper consisted of a proportional/derivative (PD) feedback loop along with conditions to maintain the horizon in the image plane.

1.3 Overview

This thesis documents the initial progress made in vision-based autonomous flight for micro air vehicles at the University of Florida. First, a relationship is derived between feature points viewed in the image plane and the aircraft equations of motion. Analytical simulations were then studied for this system to find the relation between the horizon's roll angle and the aircraft's body roll angle through camera nonlinearities. A vehicle was was then fabricated at the University of Florida equipped with a single camera for extensive autopilot testing. The algorithms for image processing and horizon detection were used from previous work done at the University of Florida's Machine Intelligence Lab and is documented in Ettinger et al. [8]. This thesis attempts to build on the works done in this particular paper by incorporating an autopilot methodology to both roll and heading using vision feedback. The control systems were designed for lateral directional using standard proportional and integral control structures. Flight test were then used to adjust the controller gains for the desired performance.

The vision feedback in this thesis uses the horizon detection algorithm to determine the roll angle of the aircraft. The main disadvantage to this algorithm restricts the aircraft to operate in an open environment where the horizon is always in view. Therefore, the work presented in this thesis is a small step in the overall vision control problem for urban environments, where the horizon is no longer in view. The detection methodology can be expanded to to features such as buildings instead of the horizon approach presented in this thesis.

CHAPTER 2
DERIVATION OF COUPLED CAMERA AND AIRCRAFT EQUATIONS OF MOTION

2.1 Problem Formulation

The goal of this chapter is to formulate and derive the coupling between the equations of motion for MAV's and the motion of feature points, viewed by cameras attached to the aircraft, in the focal plane. These equations will be derived in a general form that will include indicies for the number of cameras, k, and the number of feature points, a, extracted by from the image. An important assumption made regarding feature points is an image processing algorithm has been created to extracted critical features of the environment, such as corners of buildings, trees and so forth.

The equations will involve knowledge of each camera's mounting position and orientation relative to the aircraft's center of mass. An illustration of this formulation is depicted in Figure 2–1 which describes the kinematics of a MAV in flight with details of the k^{th} camera. The coordinate frames, or basis frames, in this derivation consist of an inertial frame, **E**, a body fixed frame, **B**, and the k^{th} camera fixed frame, $\mathbf{C^k}$, which also can be seen in Figure 2–1. Each frame is represented as a right-handed, orthonormal basis expressed as unit vectors. The Figure also contains an image, where feature points are extracted, to emphasize the application of vision for autonomous vehicles in cluttered environments for closed-loop control.

2.2 Aircraft Parameters

The aircraft equations of motion for a MAV are the standard equations which have been derived in a typical aircraft mechanics book. The important aircraft states needed for the camera motion consist of the velocity of the center of mass of the aircraft, ${}^{E}\underline{\mathrm{v}}_{c}$, with respect to the inertial frame, the angular velocity, ${}^{E}\underline{\omega}^{B}$, of the body frame with

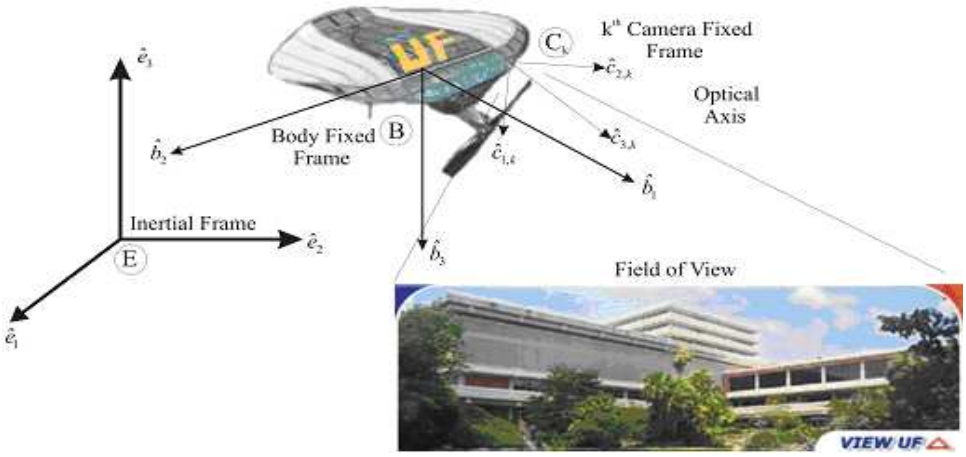


Figure 2–1: MAV Kinematics

respect to the inertial frame, the position of the center of mass of the aircraft, $\underline{R}_c$, with respect to the inertial frame, and the orientation angles of the aircraft with respect to the inertial frame. These vectors will appear explicitly in the derivation of the camera equations and are written below in terms of their components. Here the notation for a vector quantity is denoted using an under-bar, (*e.g.* $\underline{v}$ is a vector quantity).

$$^{E}\underline{v}_c = u\hat{b}_1 + v\hat{b}_2 + w\hat{b}_3$$

$$^{E}\underline{\omega}^{B} = p\hat{b}_1 + q\hat{b}_2 + r\hat{b}_3$$

$$\underline{R}_c = X_c\hat{e}_1 + Y_c\hat{e}_2 + Z_c\hat{e}_3$$

The angular velocity, $^{E}\underline{\omega}^{B}$, is a function of the time rate of change of the orientation angles. These angles correspond to a sequence of 2-D rotations which defines the orientation of the body axis relative to the inertial frame and are usually defined for an aircraft as roll, pitch, and yaw (ϕ, θ, ψ) or also referred to as a (3-2-1) Euler rotation.

2.3 Defining Camera Parameters

The parameters that will be involved in deriving the equations for a camera mounted to an aircraft include the position of the camera relative to the body basis, position of a feature point relative to an inertial basis, the position of a feature point relative to the camera basis, and the aircraft states described in the previous section.

The position vector of the k^{th} camera relative to the body basis is denoted as $\underline{\Delta}^k$ and is expressed below in component form.

$$\underline{\Delta}^k = \Delta_1^k \hat{e}_1 + \Delta_2^k \hat{e}_2 + \Delta_3^k \hat{e}_3$$

The orientation angles defined by a 3-2-1 Euler rotation matrix also relate the camera basis to the body basis in a similar fashion. This is also represented in a general form based on the number of cameras, k, to allow multiple cameras with different orientations. These angles are defined to be $(\phi^k, \theta^k, \psi^k)$ corresponding to roll, pitch, and yaw for the k^{th} Camera basis. Both the position and orientation of the k^{th} camera are illustrated in Figure 2–2.

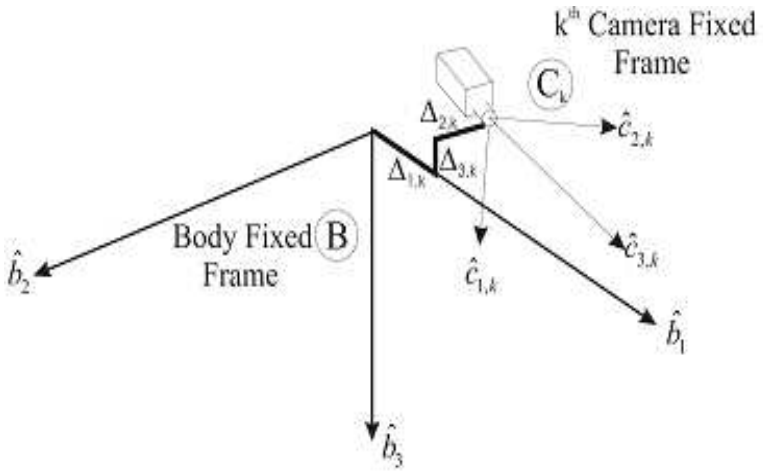


Figure 2–2: Coordinate Systems

Thus, the parameters defining the k^{th} camera described above can be assembled into a vector to simplify and compact the notation and will be referred to as α_k.

$$\underline{\alpha}_k^T(t) = \{\Delta_1^k, \Delta_2^k, \Delta_3^k, \phi^k, \theta^k, \psi^k, f^k\}$$

The parameter f^k, is the focal length of the k^{th} camera and can vary for multiple cameras. Keep in mind this vector can be extended to include radial lens distortion,

CCD array misalignment and so forth but this derivation will retain only the position, orientation, and the focal length for each camera without loss of generality.

The image captured by an on-board camera requires processing to extract feature points that are critical to the aircraft's path planning. This work has been done for many applications using pattern recognition techniques and vision processing [27]. The assumption made for this derivation is the vision processing has been addressed, extracting a minimum amount of feature points, lines, and/or planes to learn the current surroundings. A typical scenario is depicted in Figure 2–3 which shows an image of a building being reduced to feature points and lines for guidance, navigation, and control through this urban terrain. Many problems with free and occupied space still arise when a family of feature points are predetermined. Some statistical methods, based on color, in pattern recognition have attempted to address this problem with a reasonable degree of accuracy but problems arise when using this statistical model in a cluttered environment [30].

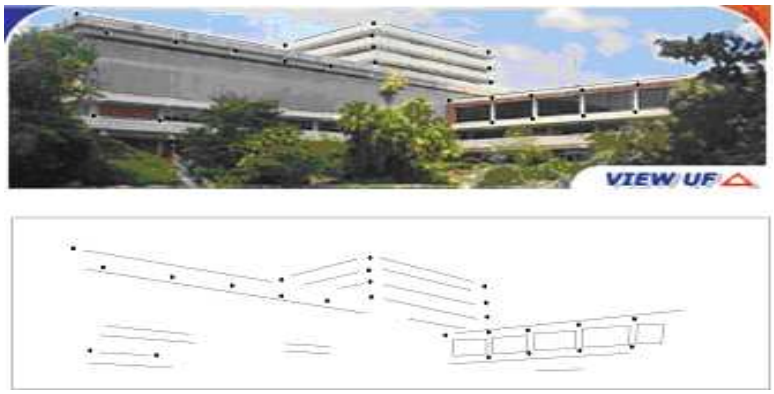


Figure 2–3: Defining Feature Points

The remaining parameters characterize the geometry between the aircraft's center of mass, the k^{th} camera's lens position and the a^{th} feature point located on points, lines and planes extracted from an image. A vector diagram of this geometry is illustrated in Figure 2–4.

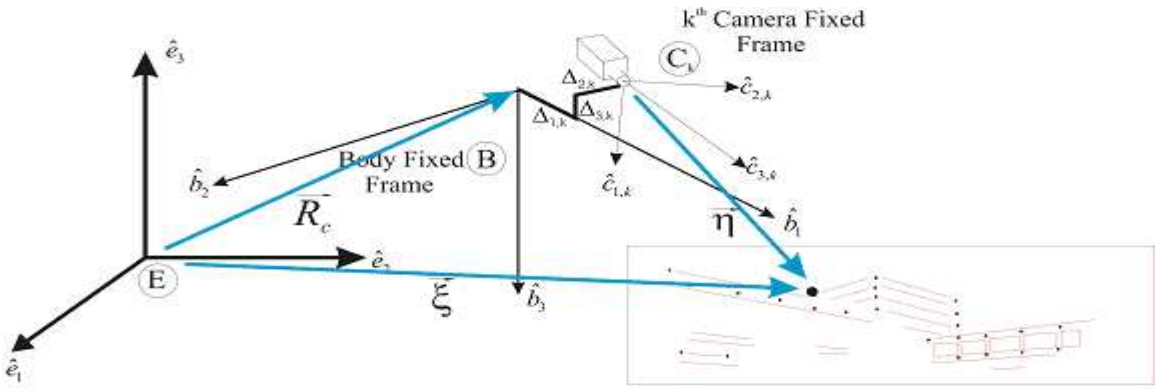


Figure 2–4: Vector Diagram

Here the position of the center of mass of the aircraft is shown and is again denoted by $\underline{R}_c$. The remaining parameters shown in this figure consist of the position of the a^{th} feature point relative to the both the inertial frame and the camera frame, denoted as $\underline{\xi}^a$ and $\underline{\eta}^a$. The components of these two vectors are expressed in the basis written below.

$$\underline{\xi}^a = \xi_1^a \hat{e}_1 + \xi_2^a \hat{e}_2 + \xi_3^a \hat{e}_3$$

$$\underline{\eta}^a = \eta_1^a \hat{c}_1 + \eta_2^a \hat{c}_2 + \eta_3^a \hat{c}_3$$

2.4 Camera Equations of Motion

Recall that the basis frames defined earlier are right-handed, orthonormal meaning the axes permutate according to the cross product and the dot product between normal unit vectors are zero. A 2D axis rotation can then be defined through the direction cosine, relating one set of coordinates to a new set which is only a rotation from the first. The 2D transformations defined below are rotations about a particular axis. The subscript defines the axis of rotation, for example $l_3(\phi)$ is a rotation about the three axis by an angle ϕ.

$$[l_3(\psi^k)] = \begin{bmatrix} c\psi^k & s\psi^k & 0 \\ -s\psi^k & c\psi^k & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad [l_2(\theta^k)] = \begin{bmatrix} c\theta^k & 0 & -s\theta^k \\ 0 & 1 & 0 \\ s\theta^k & 0 & c\theta^k \end{bmatrix} \quad [l_1(\phi^k)] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\phi^k & s\phi^k \\ 0 & -s\phi^k & c\phi^k \end{bmatrix}$$

Using a series of three 2-D rotations, a set of coordinates can be transformed uniquely into any desired orientation. Therefore, the basis transformation between body basis and the k^{th} camera basis can be defined as follows:

$$\begin{Bmatrix} \hat{c}_1 \\ \hat{c}_2 \\ \hat{c}_3 \end{Bmatrix}^k_{\hat{c}} = [l_1(\phi^k)][l_2(\theta^k)][l_3(\psi^k)] \begin{Bmatrix} \hat{b}_1 \\ \hat{b}_2 \\ \hat{b}_3 \end{Bmatrix}^k_{\hat{b}} \tag{2.1}$$

Likewise, the same sequence can be used to relate the inertial basis with the body basis through the aircraft's orientation angles.

$$\begin{Bmatrix} \hat{b}_1 \\ \hat{b}_2 \\ \hat{b}_3 \end{Bmatrix}_{\hat{b}} = [l_1(\phi)][l_2(\theta)][l_3(\psi)] \begin{Bmatrix} \hat{e}_1 \\ \hat{e}_2 \\ \hat{e}_3 \end{Bmatrix}_{\hat{e}} \tag{2.2}$$

2.4.1 Feature Point Position

The derivation of the camera equations of motion starts by summing the position vectors, illustrated in Figure 2–4, for the a^{th} feature point relative to the inertial frame.

$$\underline{\xi}^a = \underline{R}_c + \underline{\Delta}^k + \underline{\eta}^a \tag{2.3}$$

Solving Equation 2.3 for the position of the a^{th} feature point relative to the camera basis, results with the following vector equation.

$$\underline{\eta}^a = \underline{\xi}^a - \underline{R}_c - \underline{\Delta}^k \tag{2.4}$$

Each term in Equation 2.4 can be expressed in the camera basis by applying the Euler transformations, shown in Equations 2.1 and 2.2 resulting in the final vector equation for the feature position relative to the camera basis.

$$\begin{aligned}\underline{\eta}^a_{\hat{c}} = &[l_1(\phi^k)][l_2(\theta^k)][l_3(\psi^k)][l_1(\phi)][l_2(\theta)][l_3(\psi)](\underline{\xi}^a - \underline{R}_c)_{\hat{e}} \\ &- [l_1(\phi^k)][l_2(\theta^k)][l_3(\psi^k)]\underline{\Delta}^a_{\hat{b}}\end{aligned} \tag{2.5}$$

Writing this in a more compact form using indicial notation results with the the final expression of the position of the a^{th} feature point relative to the camera basis.

$$\eta^a_i = l(\phi^k,\theta^k,\psi^k)_{ij}\, l(\phi,\theta,\psi)_{js}\{\xi^a - R_c\}_s - l(\phi,\theta,\psi)_{ij}\Delta^k_j \tag{2.6}$$

2.4.2 Feature Point Velocity

We can characterize a feature point in the focal plane further by deriving it's velocity vector. The velocity of the a^{th} feature point in the focal can be found by taking the time derivative of Equation 2.4 with respect to the inertial frame.

$$\frac{^E d}{dt}(\underline{\eta}^a) = \frac{^E d}{dt}(\underline{\xi}^a) - \frac{^E d}{dt}(\underline{R}_c) - \frac{^E d}{dt}(\underline{\Delta}^k) \tag{2.7}$$

Looking at each term individually, the derivative is with respect to the inertial frame while most of these vectors are expressed in other basis. Therefore, the Derivative Theorem is employed on terms not expressed in the inertial basis. For a general vector, $\underline{A}$ expressed in the **J** frame, the Derivative Theorem states the time rate of change of $\underline{A}$ in the **I** frame is equal to the time rate of change of $\underline{A}$ in **J** plus the angular velocity of frame **J** with respect to frame **I** crossed with $\underline{A}$. Mathematically, this is shown as follows:

$$\frac{^I d}{dt}(\underline{A}) = \frac{^J d}{dt}(\underline{A}) + {}^I\underline{\omega}^J \times \underline{A}$$

The position vector, $\underline{\xi}^a$, is constant in magnitude and direction with respect to time in the inertial frame and therefore the first term on the right hand side of Equation 2.7 is zero. The position vector of the aircraft's center of mass, $\underline{R}_c$, is expressed in the inertial basis and therefore the time derivative becomes the following:

$$\frac{^{E}d}{dt}(\underline{R}_c) = \dot{\underline{R}}_c$$

Using the Derivative Theorem described above for the remaining vectors $\underline{\eta}^a$ and $\underline{\Delta}^k$ and combining results discussed for each term, Equation 2.7 becomes the following:

$$\frac{^{C}d}{dt}(\underline{\eta}^a) + {}^{E}\underline{\omega}^{C} \times \underline{\eta}^a = -\dot{\underline{R}}_c - \frac{^{B}d}{dt}(\underline{\Delta}^k) - {}^{E}\underline{\omega}^{B} \times \underline{\Delta}^k \tag{2.8}$$

This equation can be reduced further by analyzing the term $\frac{^{B}d}{dt}(\underline{\Delta}^k)$. This is the time rate of change of $\underline{\Delta}^k$ with respect to the body basis. For a fixed camera position, this derivative is zero because it's constant in magnitude and direction relative to the body basis and therefore reduces Equation 2.8 to the following

$$\frac{^{C}d}{dt}(\underline{\eta}^a) + {}^{E}\underline{\omega}^{C} \times \underline{\eta}^a = -\dot{\underline{R}}_c - {}^{E}\underline{\omega}^{B} \times \underline{\Delta}^k$$

Solving this equation for the velocity of the a^{th} feature point with respect to the camera basis is found by manipulating the above equation.

$$\frac{^{C}d}{dt}(\underline{\eta}^a) = -\dot{\underline{R}}_c - {}^{E}\underline{\omega}^{B} \times \underline{\Delta}^k - {}^{E}\underline{\omega}^{C} \times \underline{\eta}^a \tag{2.9}$$

Using the Addition Theorem, each angular velocity term can be expressed as a sum of intermediate coordinate transformations until the desired basis is reached. For example, the angular velocity of ${}^{A}\underline{\omega}^{C}$ is the sum of the angular velocity going from frames **A** to **B** and the angular velocity going from frames **B** to **C**, where frame **B** is defined as an intermediate frame. In this derivation transforming from frame **E** to **B** requires two intermediate frames to accomplish this task, where **F** and **G** are the intermediate frames for the angular velocity below.

$$\begin{aligned} {}^{E}\underline{\omega}^{B} &= {}^{E}\underline{\omega}^{F} + {}^{F}\underline{\omega}^{G} + {}^{G}\underline{\omega}^{B} \\ &= \dot{\psi}\hat{f}_3 + \dot{\theta}\hat{g}_2 + \dot{\phi}\hat{b}_1 \end{aligned}$$

Each component of this equation can be written in the **B** basis by the rotational transformations defined in Equation 2.2.

$$ {}^{E}\underline{\omega}^{B} = [l_1(\phi)][l_2(\theta)] \left\{ \begin{array}{c} 0 \\ 0 \\ \dot{\psi} \end{array} \right\}_{\hat{f}} + [l_1(\phi)] \left\{ \begin{array}{c} 0 \\ \dot{\theta} \\ 0 \end{array} \right\}_{\hat{g}} + \left\{ \begin{array}{c} \dot{\phi} \\ 0 \\ 0 \end{array} \right\}_{\hat{b}} $$

Recall, this angular velocity expressed in the **B** basis as the following

$$ {}^{E}\underline{\omega}^{B} \triangleq \left\{ \begin{array}{ccc} p & q & r \end{array} \right\}^{T}_{\hat{b}} $$

The angular velocity of the **C** frame with respect to the **E** frame requires two additional intermediate frames, **O** and **P**, from the angular velocity shown above.

$$ \begin{aligned} {}^{E}\underline{\omega}^{C} &= {}^{E}\underline{\omega}^{F} + {}^{F}\underline{\omega}^{G} + {}^{G}\underline{\omega}^{B} + {}^{B}\underline{\omega}^{O} + {}^{O}\underline{\omega}^{P} + {}^{P}\underline{\omega}^{C} \\ &= \dot{\psi}\hat{f}_3 + \dot{\theta}\hat{g}_2 + \dot{\phi}\hat{b}_1 + \dot{\psi}^k\hat{o}_3 + \dot{\theta}^k\hat{p}_2 + \dot{\phi}^k\hat{c}_1 \end{aligned} $$

Likewise, each component of this equation can be written in the **C** basis by the rotational transformations defined in Equations 2.1 and 2.2.

$$ \begin{aligned} {}^{E}\underline{\omega}^{C} = {} & [l_1(\phi^k)][l_2(\theta^k)][l_3(\psi^k)][l_1(\phi)][l_2(\theta)] \left\{ \begin{array}{c} 0 \\ 0 \\ \dot{\psi} \end{array} \right\}_{\hat{f}} + [l_1(\phi^k)][l_2(\theta^k)][l_3(\psi^k)][l_1(\phi)] \left\{ \begin{array}{c} 0 \\ \dot{\theta} \\ 0 \end{array} \right\}_{\hat{g}} \\ & + [l_1(\phi^k)][l_2(\theta^k)][l_3(\psi^k)] \left\{ \begin{array}{c} \dot{\phi} \\ 0 \\ 0 \end{array} \right\}_{\hat{b}} + [l_1(\phi^k)][l_2(\theta^k)] \left\{ \begin{array}{c} 0 \\ 0 \\ \dot{\psi}^k \end{array} \right\}_{\hat{o}} \\ & + [l_1(\phi^k)] \left\{ \begin{array}{c} 0 \\ \dot{\theta}^k \\ 0 \end{array} \right\}_{\hat{p}} + \left\{ \begin{array}{c} \dot{\phi}^k \\ 0 \\ 0 \end{array} \right\}_{\hat{c}} \end{aligned} $$

Defining the components of the equation above as $\underline{\Omega}^k$ for each camera expressed in the corresponding camera basis.

$$ {}^E\underline{\omega}^C \triangleq \underline{\Omega}^k \triangleq \left\{ \begin{array}{ccc} \Omega_x & \Omega_y & \Omega_z \end{array} \right\}^k_{\hat{c}} $$

Therefore, Equation 2.9 becomes the following:

$$ \frac{{}^C d}{dt}(\underline{\eta}^a) = -\underline{\dot{R}}_c - {}^E\underline{\omega}^B \times \underline{\Delta}^k - \underline{\Omega}^k \times \underline{\eta}^a \tag{2.10} $$

$$ \left\{ \begin{array}{c} \dot{\eta}_x \\ \dot{\eta}_y \\ \dot{\eta}_z \end{array} \right\} = - \left\{ \begin{array}{c} \dot{X}_c \\ \dot{Y}_c \\ \dot{Z}_c \end{array} \right\}_{\hat{e}} - \left\{ \begin{array}{c} q\Delta^k_z - r\Delta^k_y \\ r\Delta^k_x - p\Delta^k_z \\ p\Delta^k_y - q\Delta^k_x \end{array} \right\}_{\hat{b}} - \left\{ \begin{array}{c} \Omega^k_y\eta^a_z - \Omega^k_z\eta^a_y \\ \Omega^k_z\eta^a_x - \Omega^k_x\eta^a_z \\ \Omega^k_x\eta^a_y - \Omega^k_y\eta^a_x \end{array} \right\}_{\hat{c}} \tag{2.11} $$

Transforming all coordinate components into the C basis and using indicial notation results with a final expression for the velocity of the a^{th} relative to the k^{th} camera frame.

$$ \begin{aligned} \dot{\eta}^a_i = &-l(\phi^k,\theta^k,\psi^k)_{ij}\, l(\phi,\theta,\psi)_{js}\{\dot{R}_c\}_s - l(\phi^k,\theta^k,\psi^k)_{ij}\{\varepsilon_{iop}\, {}^E\omega^B_o \Delta^k_p\} \\ &- \varepsilon_{ilk}\, {}^E\Omega^C_l \eta^a_k \end{aligned} \tag{2.12} $$

CHAPTER 3
CAMERA NONLINEARITIES

3.1 Focal Length Parameter

3.1.1 Focal Plane Position

The coordinates defined in the focal plane are (μ, ν) which represents a two-dimensional projection viewed by the camera. Therefore, the a^{th} feature point will appear in the focal plane with the coordinates (μ^a, ν^a) from the component equation listed below.

$$\mu^a = f\frac{\eta_x{}^a - c_x}{\eta_z{}^a - c_z} \qquad \nu^a = f\frac{\eta_y{}^a - c_y}{\eta_z{}^a - c_z} \tag{3.1}$$

Again, the parameter f^k is the focal length of the k^{th} camera. The vector components (c_x, c_y, c_z), defined as $\underline{c}$, are the offset distances of the lens relative to the camera basis. For example, c_x is the distance the lens is offset in the x-direction from the camera basis. If the origin of the camera basis is place at the lens, Equation 3.1 reduces to Equation 3.2 (i.e. $\underline{c} = 0$).

$$\mu^a = f\frac{\eta_x{}^a}{\eta_z{}^a} \qquad \nu^a = f\frac{\eta_y{}^a}{\eta_z{}^a} \tag{3.2}$$

For simplicity, it is assumed for the remainder of the derivation the lens location and camera basis coincide and therefore Equation 3.2 will be used. This equation is only a function of the position relative to the camera basis to a feature point, derived in Chapter 2.4.1, and the camera's focal length.

In summary, the position of each feature point in space can now be characterized by it's position in the focal plane by substituting the required components of Equation 2.6 into Equation 3.2.

A simple experiment was done using Equation 3.2 to characterize the mapping associated with feature points in the inertial frame transforming to the focal plane. A pattern of feature points was defined in the inertial frame as a plane perpendicular to the Y axis and is shown in Figure 3–1. The feature points on this plane were then rotated several times to create a series of planes. Each plane was fixed on the left side while rotating the right side closer to the camera's position. An illustration of this setup is depicted in Figure 3–2, where point C is defined as the camera's position and is given as $C = \left\{ \begin{array}{ccc} 0 & 0 & 100 \end{array} \right\}^T$.

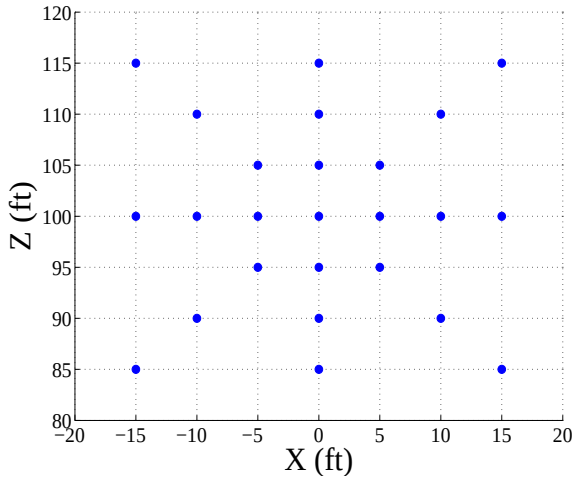
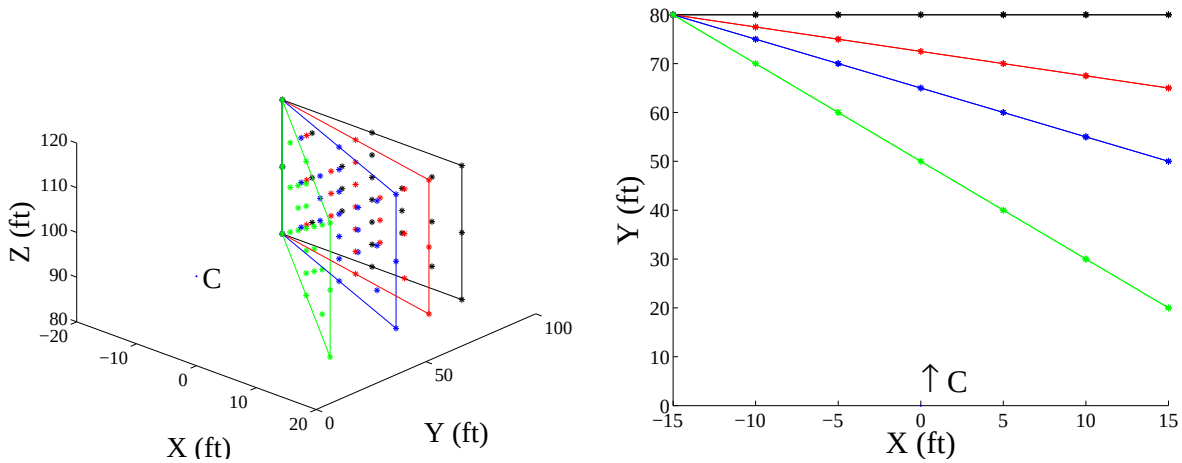


Figure 3–1: Feature Point Grid

Each plane contains this set of grid points which are then substituted into Equation 2.6, assuming a fixed aircraft with a single camera placed at the center of mass along with fixed basis frames. The feature point positions were then determined relative to the camera basis. Therefore, the following aircraft and camera parameters reduce to the following:



(a) 3D (b) XY Plane

Figure 3–2: Grid Rotation

$$
{}^{E}\underline{\mathrm{v}}_{c} = \left\{ \begin{array}{ccc} 0 & 0 & 0 \end{array} \right\}^{T}
$$

$$
{}^{E}\underline{\omega}^{B} = \left\{ \begin{array}{ccc} 0 & 0 & 0 \end{array} \right\}^{T}
$$

$$
\underline{\mathrm{R}}_{c} = \left\{ \begin{array}{ccc} 0 & 0 & 100 \end{array} \right\}^{T}
$$

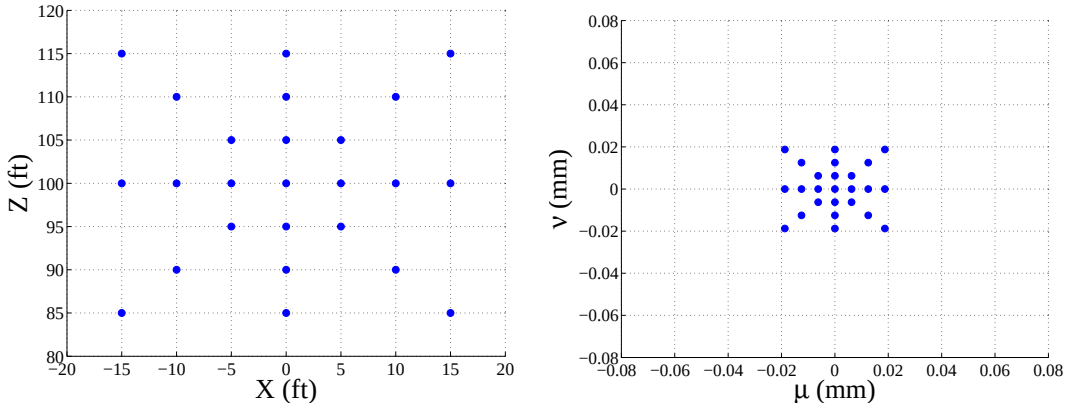
$$
\underline{\Delta} = \left\{ \begin{array}{ccc} 0 & 0 & 0 \end{array} \right\}^{T}
$$

Also, the rotation matrices reduce to the following for a level camera and MAV:

$$
[l_1(\phi^k)][l_2(\theta^k)][l_3(\psi^k)] = \begin{bmatrix} 0 & 0 & 1 \\ 0 & -1 & 0 \\ 1 & 0 & 0 \end{bmatrix}
$$

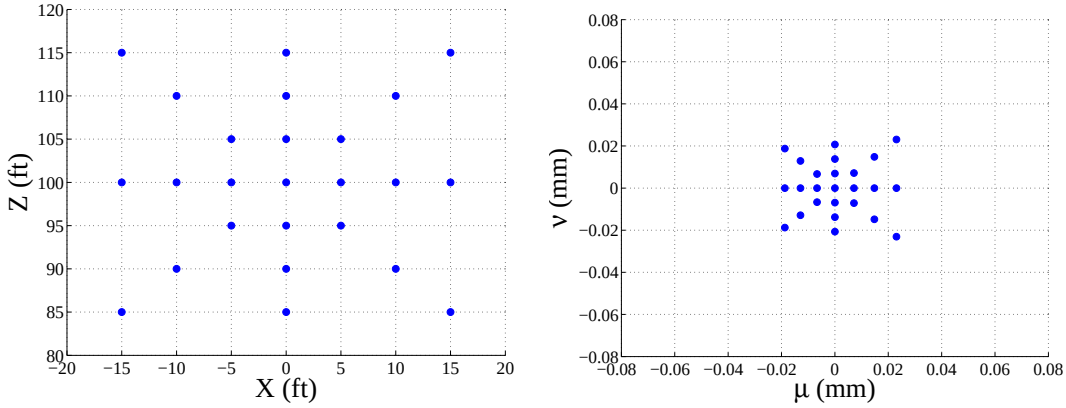
$$
[l_1(\phi)][l_2(\theta)][l_3(\psi)] = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & -1 \end{bmatrix}
$$

After the position of each feature point, η^a, was determined relative to the camera basis, the focal plane position was then calculated using Equation 3.2 through direct substitution. Figures 3–3 - 3–6 show the feature point position for each grid plane as it's rotated toward the camera's position for a set focal length, where the plane in the first set of plots is parallel to the inertial $X-Z$ plane. Here the focal length, f, was set to a constant value of $f = 0.1\ mm$ for all the plots below.



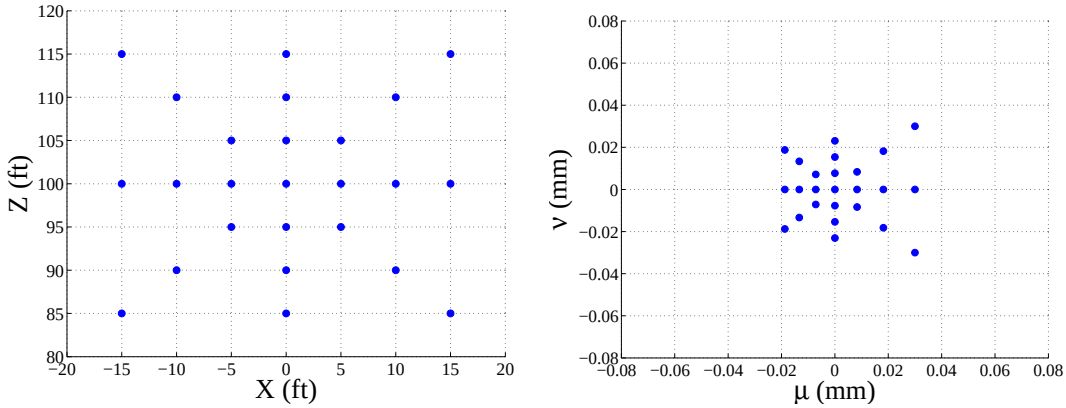
(a) Inertial Frame (b) Focal Plane

Figure 3–3: Planar Feature Points



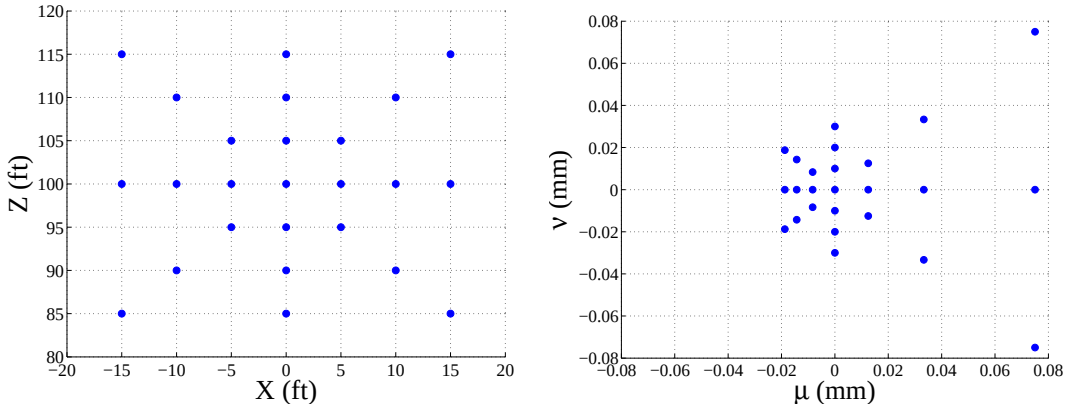
(a) Inertial Frame (b) Focal Plane

Figure 3–4: Skewed Feature Points Sloping 2.5 Units



(a) Inertial Frame (b) Focal Plane

Figure 3–5: Skewed Feature Points Sloping 5 Units



(a) Inertial Frame (b) Focal Plane

Figure 3–6: Skewed Feature Points Sloping 10 Units

Notice as each plane is rotated toward the camera, the focal plane image becomes increasingly distorted. The first set of plots reveals when the depth, inertial Y direction, is held constant the focal plane mapping retains the the initial shape with some scaling factor proportional to the focal length. As the plane is rotated by changing the depth, the feature points which are closer appear to be spreading further apart while points that are at a greater distance appear to converge closer together. As seen in Figure 3–6, the feature points contained in the right half plane have increased their

separation distance significantly compared to the previous cases. Each grid plane rotation was doubled so a comparison can be made with the distortion distance. As the plots illustrate, as the rotation plane is doubled the resulting distortion distance has increased more than twice. Therefore, the focal length relation between feature point location and depth is nonlinear. A formal mathematical statement is presented below in Equation 3.3.

$$(\mu, \nu) = G(f^k, \xi_y^a) \tag{3.3}$$

The function G describes some nonlinear function which maps feature point position in inertial space to the focal plane. This function only relates the nonlinearities describing depth perception onto the image plane through focal length and is strongly a function of the ξ_y component.

3.1.2 Focal Plane Velocity

The final step is to formulate an expression for the velocity of feature points in the focal plane. Therefore, the focal plane velocity is found by taking the time derivative of the position defined in Equation 3.2.

$$\dot{\mu}^a = f\frac{\dot{\eta}_x^a}{\eta_z^a} - f\frac{\eta_x^a}{(\eta_z^a)^2}\dot{\eta}_z^a \qquad\qquad \dot{\nu}^a = f\frac{\dot{\eta}_y^a}{\eta_z^a} - f\frac{\eta_y^a}{(\eta_z^a)^2}\dot{\eta}_z^a \tag{3.4}$$

The focal plane velocity of the a^{th} feature point relative to the k^{th} is finalized by substituting both equations for position and velocity derived in Equation 2.6 and 2.12 into Equation 3.4

3.2 Lens Distortion

Another important nonlinearity associated with a camera is image distortion due to the curvature of the lens. This effect can be described as a tunnel vision image, where feature points on the extreme sides are drawn toward the horizon centerline. This is a

planar distortion unlike the the focal length mapping described in the previous section in which the shape was unchanged for the planar case.

An experiment was conducted using a standard Marshell CMOS color camera with 310 TV lines of resolution to characterize image distortion due to lens curvature. Figure 3–7 illustrates the dramatic warping done to a set of evenly spaced grid points.

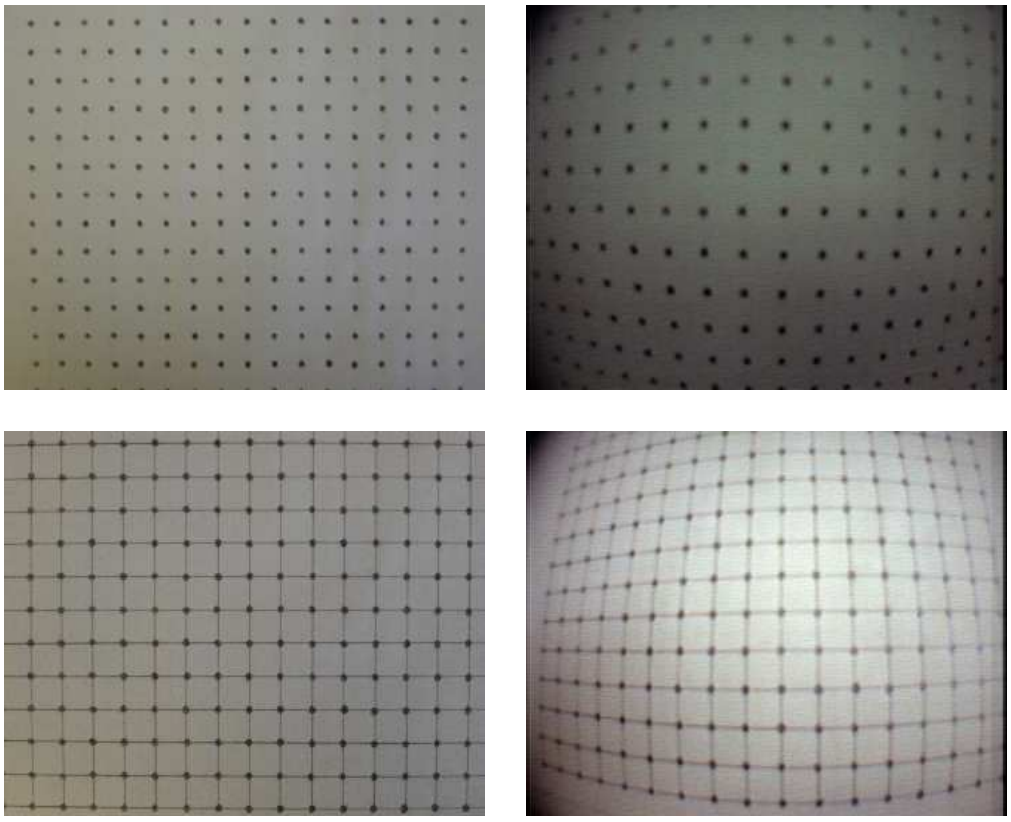


Figure 3–7: Lens Curvature

The horizontal and vertical centerlines retain a linear curve while lines above and below follow this parabolic concave and convex mapping that is axisymmetric. Likewise, this mapping can be expressed mathematically as a nonlinear function depending on inertial space and the focal length, as given in Equation 3.5.

$$(\mu, \nu) = L(f^k, \xi_x^a, \xi_z^a) \tag{3.5}$$

The camera nonlinearities described above will have a significant effect on a controller design for obstacle detection and avoidance. Buildings appearing in the image plane, depending on their location, will appear closer or further away than they actually are due these distortions. Therefore, the functions G and L are required to accurately describe the motion of feature points in real space for control designs and will vary for each particular camera.

CHAPTER 4
LATERAL AUTOPILOT CONTROL DESIGN

The long term goal of this autopilot is to structure the controls system that would allocate GPS waypoint tracking for MAVs using vision feedback. The controls layout for an aircraft is typically separated into lateral directional and longitudinal controllers. This section will document the design of a lateral autopilot control system for a MAV by continuing the investigation of vision-based antonomy presented in the literature [8, 18]. This design can be easily extended to lateral waypoint tracking by incorporating an outer-loop that involves guidance algorithms.

The lateral control system deals with maintaining a desired roll and/or heading angle through aileron and rudder deflections. By designing the lateral controls system independently, some coupling in longitudinal states will result from these surface deflections. For example, the sideslip induced by a roll angle can have a significant on the longitudinal states with the loss of altitude. This may be modified in future works by considering an aileron-rudder interconnect to prevent the nose from drifting downward during a bank hold. This thesis will only consider the effects directly related to the lateral states and will disregard these coupling terms.

The only lateral directional terms that are of interest for this control design include roll and heading angle. For this section, these measurements are assumed to be known perfectly. The sensors which provide these quantities will introduce a finite sampling rate along with some uncertainty, which will be discussed later in the chapter and again in Chapter 6.1.

The control architecture for this autopilot will begin with an inner-loop controller and continue to an outer-loop design. The inner-loop controller will attempt to maintain a desired roll angle, while the outer-loop will control the heading direction.

The general designs of these controls system will be discussed in further detail during the remaining sections.

4.1 Roll Control Design

A standard approach using proportional and integrator blocks was chosen for this particular loop. This classical controls technique computes an error based on the difference between the desired and measured state and makes that proportional to a servo deflection. The integrator block helps to manipulate both the steady state error and the rate of the response. The lateral steady state error correction was used to account for asymmetries in the airplane due to construction and any net torques caused by the motor.

For the case of a roll controller, the architecture can be accurately described in block diagram form in Figure 4–1.

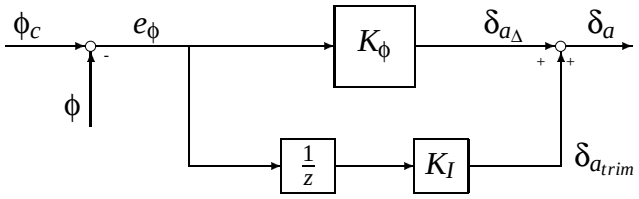


Figure 4–1: Roll Control Autopilot

The trim control block consist of a discrete time integral, and a proportional gain, K_I. The integral block is used primarily for improving steady state by continuously driving the roll error to zero. Therefore, for a zero degree roll command the control will maintain straight and level flight. The proportional gain is then applied to the integral state to convert this value to a trim deflection, $\delta_{a_{trim}}$, which is then sent to the servos.

The remaining control element is a proportional gain where the difference between the measured roll, ϕ, and the roll command, ϕ_c is defined as the error. This roll error is then converted to an aileron deflection, δ_{a_Δ}, by the proportional gain, K_ϕ. The total

aileron deflection is then computed by summing the individual deflections determined from both loops, as shown in Equation 4.1.

$$\delta_a = \delta_{a_{trim}} + \delta_{a_\Delta} \tag{4.1}$$

This controls system is only considering maintaining roll angle to a desired value and does not incorporate coupling terms during this maneuver, such as maintaining altitude during a bank hold.

4.2 Heading Control Design

For the heading control design, a nonzero roll angle will be used to change the measured heading direction. Therefore, the roll controller designed in the previous section will be implemented in the heading controller as the inner loop. The controls structure is kept consistent by using a proportional feedback controller for heading as well. The block diagram for this open-loop controls system is depicted in Figure 4–2.

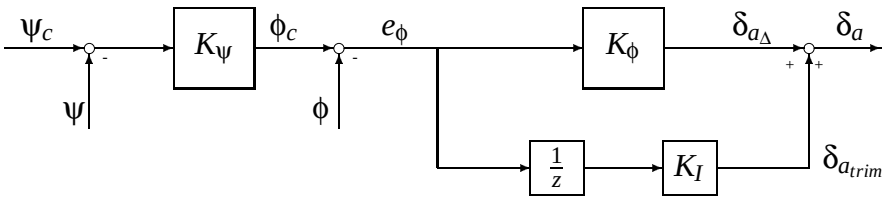


Figure 4–2: Heading Control Autopilot

The heading error is computed and set proportional to a roll command through K_ψ. As the aircraft tracks a roll angle, the heading will begin to change which in turn will reduce the heading error. So effectively, as the measured heading approaches the commanded heading the error will go to zero and the MAV will begin to roll back and eventually reach 0 *deg* roll angle. The aileron deflections are again computed using Equation 4.1.

4.3 Closed-Loop Control

4.3.1 Bank Hold

Careful consideration of the vision-based sensor should be taken before designing the closed-loop roll control system. The resolution of the measured roll angle was found to be a constant ± 4.45 *deg*. Potential problems will exist if the roll error is computed and used in the control loop. For example, if the roll command was 2 *deg* the error in roll angle would always be a nonzero value, due to the fact that the vision system is unable to resolve below 4.45 *deg*, always resulting with a nonzero aileron command. Physically the airplane will continuously try to drive the error to zero by deflecting in the correct direction causing an overshooting to the measured resolution followed by an opposite deflection to correct for the overshoot. This type of response can be described as a limit cycle oscillation centered around the desired command.

The roll command is replaced with discrete values between the range $-40 < \phi_c < 40$ *deg*, where the maximum roll command was set to ± 40 *deg*, to avoid this resolution issue. A bench test was then done to verify all possible values the roll sensor would measure between 0 and ± 40 *deg* and is documented in Table 4–1.

Table 4–1: Discrete Roll Command Output Values

Roll Command Range (*deg*)	Discrete Roll Command (*deg*)	Roll Command Range (*deg*)	Discrete Roll Command (*deg*)
$0 \leqslant \phi_c < 2.2$	0	$-2.2 \leqslant \phi_c < 0$	0
$2.2 \leqslant \phi_c < 6.6$	4.45	$-6.6 \leqslant \phi_c < -2.2$	-4.45
$6.6 \leqslant \phi_c < 11.1$	8.90	$-11.1 \leqslant \phi_c < -6.6$	-8.90
$11.1 \leqslant \phi_c < 15.5$	13.35	$-15.5 \leqslant \phi_c < -11.1$	-13.35
$15.5 \leqslant \phi_c < 20.0$	17.80	$-20.0 \leqslant \phi_c < -15.1$	-17.80
$20.0 \leqslant \phi_c < 24.4$	22.25	$-24.4 \leqslant \phi_c < -20.0$	-22.25
$24.4 \leqslant \phi_c < 28.9$	26.70	$-28.9 \leqslant \phi_c < -24.4$	-26.70
$28.9 \leqslant \phi_c < 33.3$	31.15	$-33.3 \leqslant \phi_c < -28.9$	-31.15
$33.3 \leqslant \phi_c < 37.0$	35.60	$-37.0 \leqslant \phi_c < -33.3$	-35.60

Another potential resolution problem occurs when computing the roll error for the integration block. For the example given above, the value of the error will alternate

between 2 when the measured roll angle is zero and 2.45 when the measured roll is angle is 4.45; again creating a possible limit cycle oscillation due a nonzero roll error. Therefore, when the roll error is less than the resolution for a given command, the error is then set to zero.

The closed-loop design of the roll controller can be described in block diagram form through Figure 4–3. Some additional blocks were added to account for these uncertainties in the system and camera nonlinearities.

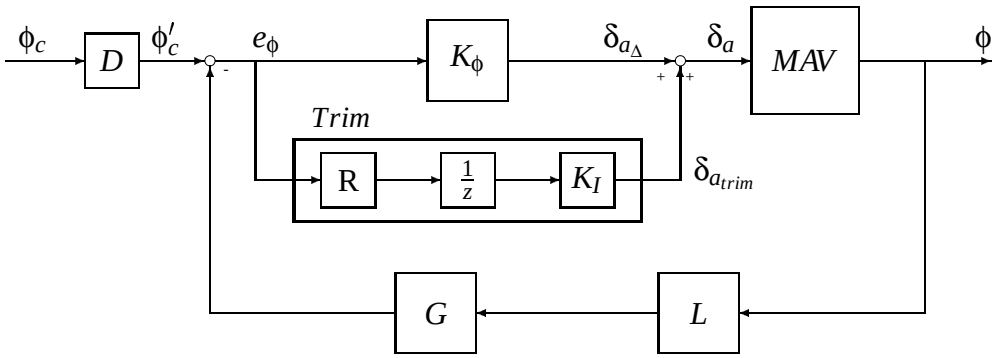


Figure 4–3: Closed-Loop Roll Control Design

The discretizer blocks, *D* and *R*, were used to eliminate limit cycle oscillations associated with a low resolution in roll angle. Block *D* converts the given roll command into discrete values associated with the horizon sensor, where these values are given in Table 4–1. The error associated with the horizon resolution, as described above, is represented as block *R*. The remaining blocks *G* and *L* describe the nonlinear mapping of feature points in inertial space to the focal plane due to depth and lens distortion, which is discussed in Chapter 3.

4.3.2 Heading Hold

As stated before, the heading controller consists of an outer-loop feedback using yaw angle to command the inner-loop bank hold. The closed-loop block diagram for a proportional heading controller is depicted in Figure 4–4.

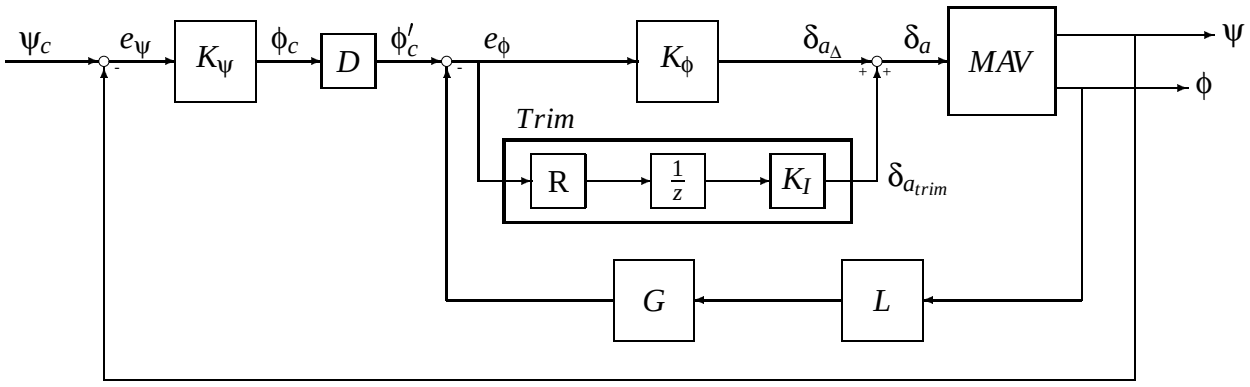


Figure 4–4: Closed-Loop Heading Control Design

The lateral control system using the roll control will be applied to a particular MAV testbed, described in the Chapter 5.1, using the horizon detection algorithm as a sensor to estimate roll. The gains will be determined through numerous flight tests and documented in the remaining sections. These results along with the final response plots will be presented in Chapter 6.2.

CHAPTER 5
EXPERIMENTAL SETUP

5.1 Micro Air Vehicle Description

As stated in Chapter 1.2.1, the University of Florida has an on campus facility to fabricate MAVs in various sizes customized for a particular mission. In each case, the volume of the fuselage is designed to house flight components and payload necessary for a particular mission. In the case of an vision-based autopilot MAV, data and video packages along with the required batteries must be incorporated in the design of the fuselage. Once the total weight of the vehicle is determined the required wing area and propulsion system can be selected.

The MAV build for this particular mission was a modified design of the standard MAV at UF. The structural material of the airframe was still constructed using layers of composite carbon fiber attached to a flexible wing design, as described in Chapter 1.2.1. A series of pictures of this MAV are shown in Figure 5–1. Notice the location of the camera has replaced the traditional position of a propeller design. Some advantages of this camera position are (1) a direct correlation between horizon roll angle and body axis roll and (2) a camera pointing out the nose will acquire better images due to the steady flow conditions. The propeller is then placed behind the wing on a shaft that connects the fuselage to the horizontal and vertical stabilizers. This type of propulsion system allows for a more steady laminar flow around the wing due to the absence of shed vortices from the propeller.

The final design of the MAV testbed used for vision-based autopilot control has typical properties listed in Table 5–1.

A wing morphing technique was applied to the flexible wings to effectively control the roll state. The term morphing, was coined because of the bending or twisting of

Figure 5–1: MAV Prototype for Vision-Based Control

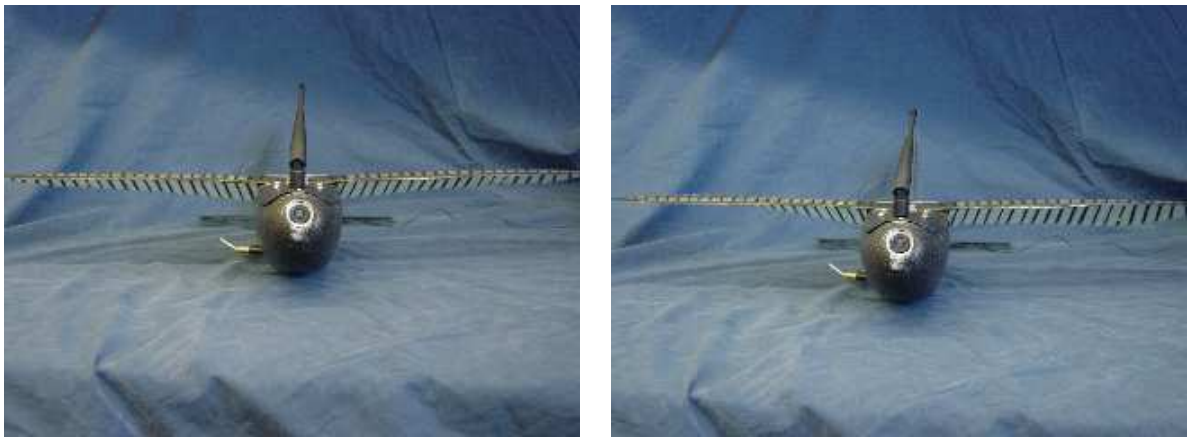
Table 5–1: MAV Properties

Wingspan	24 *in*
Weight	500 *grams*
Payload Capacity	200 *grams*
Actuators	4

the wings to change camber, cord, span, area, *etc*, which effectively changes the flight characteristics of the vehicle through angle of attack. This was first implemented mainly because MAV's with these types of flexible wings are difficult to incorporate a standard aileron, due to additional structural reinforcements needed to divide the wing. After extensive flight test, the morphing of the wing demonstrated a substantial control authority in roll with little induced yaw (i.e. a more pure roll response). The University of Florida has been looking at several simple ways to morph the wing for roll control [12], one in particular was used in the design of this MAV. This design used a torque rod mounted near the wing tip and connected to a servo inside the fuselage which morphed the wing by pushing or pulling on the rod. The implementation of this concept on the current MAV is illustrated in Figure 5–2.

Figure 5–2: Torque Rod Design

An illustration in Figure 5–3 shows the slight change in effective wing area by comparing the undeflected wing to the morphing deflection. For this particular example, the left wing is deflected downward which in turn produces a positive roll (i.e. right wing down) by the additional lift on the left wing. Flight test have shown that morphing only requires a small amount of servo deflection to produce reasonable bank angles.



(a) Wings Undeflected (b) Wings Deflected using Morphing

Figure 5–3: Morphing Control Effectors

The remaining control surfaces, elevator and rudder, were implemented in more traditional way using the basic elevator on the horizontal stabilizer and rudder on the vertical stabilizer. These surfaces along with the morphing deflection constitutes full

rotational degrees of freedom for the MAV. The servo connectors to both the elevator and rudder were placed through a concentric metal pipe which passes in the middle of the propeller bearings. This design provides a compact method which helps to conceal these connectors mainly from being damaged. The servos used in this MAV were model DS 281 made by JR Servo. The control surfaces required four servos; two for the morphing and one for each elevator and rudder, which all weighs approximately 80 *grams*.

The propulsion system is powered through a Hacker brushless electric motor that spins roughly at 60,000 *RPM* and can produce a maximum of 16 *oz* of thrust. The motor is attached to a 6 *in* propeller through a two stage gearing system, which is shown in Figure 5–4. Considering the weight of the vehicle and the on board power, flight durations usually last approximately 15 *min* at full throttle using a single 3 cell Lithium battery. This 3 cell battery also powers the servos through a transceiver board.

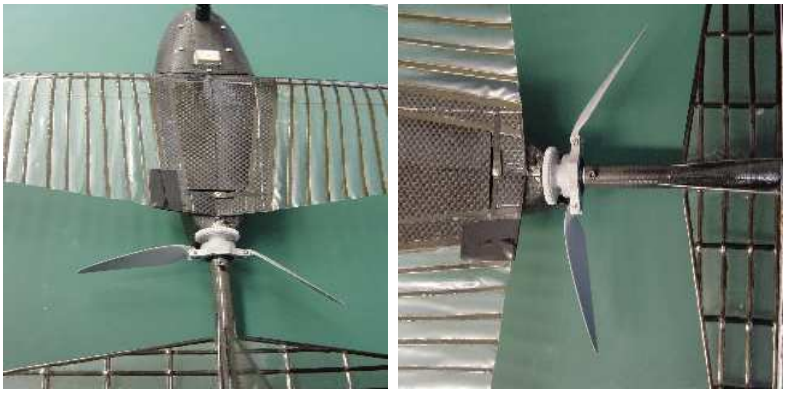


Figure 5–4: Push Tail Propeller Design

The on board sensors include GPS for location, speed, and course, an altimeter for altitude, and the vision system for horizon detection. The GPS sensor, manufactured by Furuno, uses the current and last position to calculate both speed and course with an accuracy of 5 *deg*, where the position itself is accurate to 20 *ft*. The output units consist of longitude and latitude in degrees, course in degrees, and the speed in *ft/s*. The altimeter is a pressure based sensor which measures the relative gage pressure and

has a resolution of 20 *feet* when converted to altitude. The camera of the vision system is a Marshall color CMOS camera with a resolution of 310 TV lines. The camera introduces a nonlinear mapping of feature onto the image plane due to the focal length and the curvature of the lens, which was described in Chapter 3.

5.2 Hardware Architecture

The components used to demonstrate vision control was restricted to a 200 *gram* on-board payload with a size constraint of 3 *inches*. Therefore, it was decided for the initial platform to transfer all data processing to a ground station laptop. This also allowed for human monitoring of the data on the ground during flight test. The main hardware components consist of an on-board computer networked through a wireless data-link, transceivers, a ground station laptop, a Sony Video Walkman, and USB converters. An illustration of the system architecture is depicted in Figure 5–5. The red and the black transmission lines indicate the video and sensor data steams correspondingly.

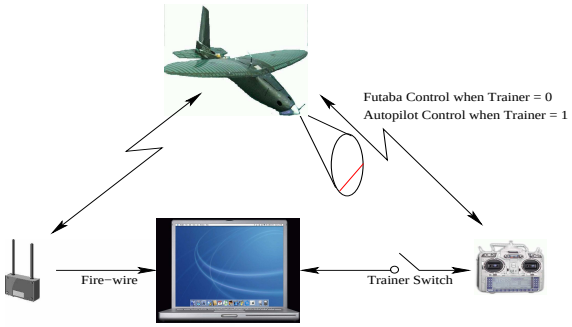


Figure 5–5: Hardware Architecture

Here the MAV contains an on-board custom microcontroller-based computer, which communicates all sensor and video data to the ground station. This micro-controller is a modified version of an Ateml AVR Mega128. An additional board was developed to increase the amount of Flash memory of the AVR Mega128 from 128kB to 256MB through an optional attachment and was used for data logging. The

complete computer system is 2-inches by 1.5 inches by 0.5 inches and weighs approximately 36 *grams*. The video and sensor data is collected through the this on-board computer and streamed downed to the ground station for data processing. The ground station, after processing this data, closes the control loop by sending servo commands back to the MAV. Additional data, such as servo command and deflection, was also sent to the ground station through the microcontroller to verify the control design.

The sensor data and video streams are broadcast ed through separate transmission frequencies. The sensor data transceiver operates at 900MHz, providing rates up to 57.6kbps and is connected to the ground station through a UART based serial port. This transceiver was a model AC4490 made by Aerocomm. Meanwhile, the video stream operates at 2.4GHz and is interfaced to the ground station through a Sony Video Walkman, via firewire.

For the initial flight test stages, a pilot was kept in the control loop to provide an override capability during takeoff, landing, and autopilot recovery. This system was incorporated by rerouting the control commands sent to the MAV through a standard Futaba controller using a custom interface. This interface employed a trainer function which allowed switching between autopilot and human control instantaneously, and is also depicted in Figure 5–5. A custom board was made for this communication link to pulse-width modulate the control signal sent from the autopilot. The serial connectors for both data communication and Futaba control were converted to USB through a Keyspan adapter, which allowed easy plug and play options for the ground station.

The ground station laptop consisted of a Apple 12-inch Powerbook running at 1GHz with 512MB of RAM. The video processing on the laptop was implemented in real-time using the described horizon detection algorithm. The video firewire communication was then coded in software to couple the on-line flight image with the resulting horizon line. This was mainly done so the ground station can provide

a visual image to analyze during autopilot testing, which became a critical tool for development.

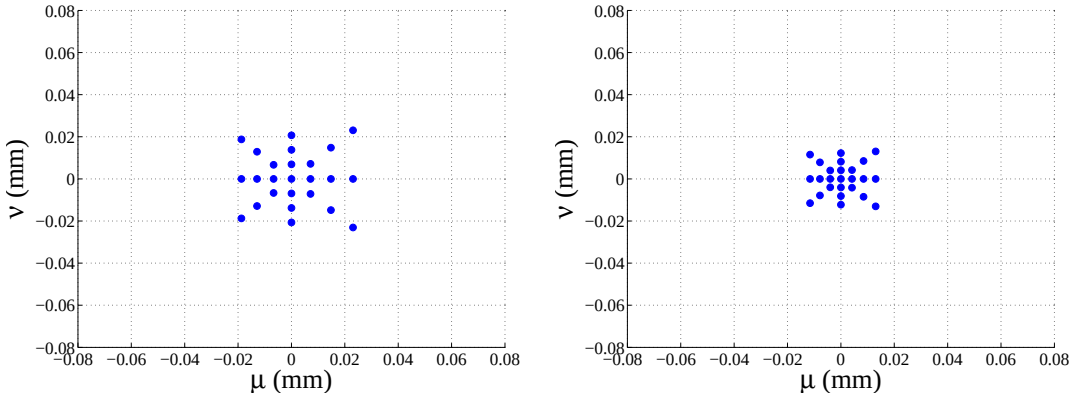
5.3 Camera Model for Horizon

The camera nonlinearities described in Chapter 3, will have a strong effect on the controller design using obstacle detection for avoidance. Buildings viewed by the camera will be warped in the image plane resulting in false representation of distance. For example, a building could appear further away in the image plane causing the controller to delay commands for avoiding this object until it's too late. Therefore, for a clutter environment, the functions G and L need to be incorporated in the control design.

In the case of horizon detection, simple analysis can validate some assumptions for these functions. The plots in Figures 3–3 and 3–4 revealed that (1) if the depth of an object is constant the original shape is maintained in the image plane and (2) a small rotation to change the depth results with a small distortion. Therefore, if the distance between the object and the camera is increased to where it's much larger than the distance in depth, the object's shape is retained in the image plane. Figure 5–6, demonstrates this by increasing the relative distance from the camera to the grid points with the same rotation.

The relative distance from the camera to the horizon line is much larger than the depth of the horizon and therefore can be approximated as a planar line in 3D space. This approximates the G function to a linear mapping from inertial space to the image plane.

The lens distortion due to curvature still remains a nonlinearity in the system and is depicted in Figure 5–7 for a horizon example. The image on the left is of a MAV in flight at straight and level and contains the horizon roughly in the center, which correlates to little distortion. The image on the right now shows a MAV in banked flight which lowers and rises the horizon line in the image. The resulting horizon has



(a) 70 *ft* (b) 120 *ft*

Figure 5–6: Image Plane Depth Comparison

warping effects associated with both sides. Depending if the horizon is above or below the centerline, the warping will be either concave or convex. With the horizon at such a far distance the warping has little effect until the sides are approached.

Figure 5–7: Lens Curvature for a Horizon

A better study of the horizon detection algorithm may reveal a similar simplification for the function L. The horizon line is determined by characterizing each pixel to a known sky or ground color distribution. For example, shades of blue would be a common feature for the sky while shades of green and brown would model the ground. Some structure is added to the algorithm that checks the neighboring pixels to add robustness. This accounts for any misclassified shades of blue in the ground or greens and browns in the sky. A decision boundary is the found when the pixel distribution is shifted from sky to ground in the image and is approximated by the linear line through statistical regression. This linear line represents the determine horizon from which

roll and pitch percentage are calculated. Figure 5–8 shows an overlay of the horizon algorithm onto the images given Figure 5–7.

Figure 5–8: Lens Curvature with Horizon Approximation

When applying a linear regression curve fit, the warped ends of the horizon are statistical outliers compared to the overall distribution and are not represented. The resulting linear fit therefore, loses this nonlinearity and approximations can be made to neglect this distortion for this particular case.

CHAPTER 6
CLOSED-LOOP LATERAL CONTROL

6.1 Controller Implementation

This section will interface the controls system with the hardware described in Chapter 5. The architecture describes a close interaction between RC and autopilot modes. The ground station code allowed easy implementation of the controller by summing the determined control deflections around a trim condition. The commands are then over-written when the signal is transfered to RC mode, giving the pilot complete control of the MAV.

This transition from RC mode to the autopilot requires some consistent deflection during this period. An offset deflection was then coded for the initial servo positions which matched the RC trim. This offset helped remove any transition bias sent to the servos when the autopilot is activated. Therefore, Equation 4.1 now becomes Equation 6.1, with the additional offset deflection.

$$\delta_a = \delta_{a_{trim}} + \delta_{a_\Delta} + \delta_{a_{offset}} \tag{6.1}$$

The on-board sensors that provide the roll and heading angles include the horizon detection algorithm and GPS. The horizon detection calculates the roll angle made by the horizon in the image plane and is updated at 35 *Hz*. Meanwhile, the GPS sensor determines the course heading angle, relative to North, from past and present GPS location and is available at 1 *Hz*. At this sample rate, the aircraft will potentially exhibit oscillations around a given heading command.

In Chapter 5.3, the nonlinearities associated with the camera were assumed to have little effect on the horizon because of the relative distance in depth. With

this information, the roll angle determined from the horizon detection sensor can be approximated as the MAV's body axis roll angle and is used directly in feedback. The control systems for both roll and heading commands now reduce to the block diagrams shown in Figures 6–1 and 6–2.

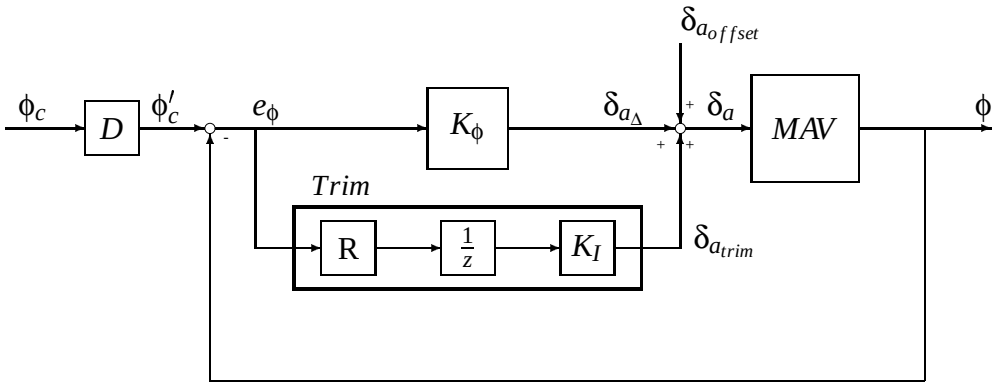


Figure 6–1: Roll Control Block Diagram

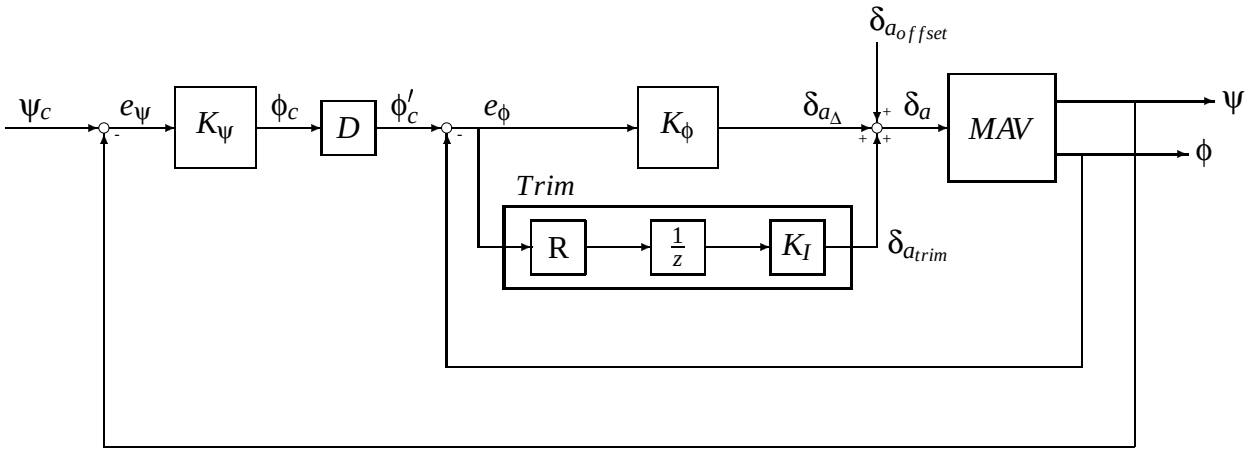


Figure 6–2: Heading Control Block Diagram

The sample rate of this outer loop is 1 *Hz*, in Figure 6–2, causing the performance of the heading tracker to diminish. For fast moving vehicle, the low sampling rate will cause overshoot oscillations in the heading response due to a low time constant. The response of the MAV during a heading command should be expected to fall within a threshold value above and below the required heading, even when the controller is

designed properly. Commercial GPS sensor are also known to have a large degree of uncertainty with some reliability issues. Some issues include interference or even loss of satellite coverage. With the loss of data due to dropouts, the performance of the controller can decrease tremendously. Drifting off course and large sudden maneuvers will result due to these types of dropouts.

6.2 Flight Testing and Results

6.2.1 Gain Tuning

As mentioned earlier, a limited amount of MAV models have been generated for control synthesis so little is known for implementation. Typically, when a dynamic model is known of the aircraft, a rough estimation of the gains can be made using a theoretical approach such as root locus. Without a model, as in this case, flight test iterations are required to adjust the gains for the desired response.

For the roll controller, two gains need to be determined, the proportional gain and the integral gain, K_ϕ and K_I. For the first few flights, these gains were set very small to effectively eliminate the controller. Once the trim of the aircraft was found for straight and level flight, using RC mode, adjustments were made to the offset deflections for autopilot transition. When a rough estimate of trim is then found for the autopilot, the gains in the controls systems are increased .

The first controller gain that was increased was the proportional gain. This gain determines the transient portion of the controller by rolling the MAV to within a steady state offset of the command. Some interesting responses occurred while increasing this gain. When the gain was too high a noticeable oscillation about the command was observed. This oscillation could have been cause by several different factors. The gain could be too large resulting in a low damping ratio in Dutch roll for the closed-loop system. The other factor under consideration is the wind gust. A MAV traveling at 25 *mph* can experience wind gust on the same order, which would drastically change the flight characteristics and most importantly the trim conditions. By decreasing this

gain, the response of the MAV would maintain a bank angle. When this gain was slightly large the bank angle response would settle down above the desired command. Likewise, when the gain was slightly lower than the final value, the response would center around a value less than the desired command.

Final flight iterations were done to acquire a response that is centered about the desired roll angle with some steady state offset. The gain is then determined through more flight iterations, where the response of the aircraft was used to determine the direction in which the gain should be changed. When the response oscillates or overshoots to a larger roll angle the gain should be decreased and increased when the response has some undershoot and settles to a smaller roll angle. Finally, the proportional gain that achieved a response centered around the desired command was determined to be $K_{\phi} = 0.75$.

The integral gain was then increased to make small corrections around the desired command for steady state conditions. Some aggressive maneuvers were observed when the integral gain was set to a large value. This integral gain is proportional to the integral of roll error, so, when the MAV was straight and level and a 0 *deg* roll command is given, a high integral gain has a smaller effect on the response because of the small error. Although, the aggressive maneuvers occurred when the MAV was placed at an initial bank angle, making the roll error large and the integral to roll error increase, when the autopilot is activated the resulting aileron deflection is large. Wind gust also had an effect on the integral loop by creating oscillations to correct for the offset. When this gain was large the wind gust would be corrected by a large deflection causing an overshoot in the opposite direction resulting in a correction for the overshoot. This behavior describes a limit cycle oscillation caused by the controller interacting with the physical flow. Some additional blocks or other control methods can be used for disturbance rejection from wind gust.

The integral gain also helps limit the rate at which the maneuver is performed. A slow response with some delay occurred when the gain was set too small. Therefore, by slowly increasing this gain the desired rate and steady state condition can be acquired. The integral gain that generated a reasonable response to commands was determined to be $K_I = 0.19$.

With these gains coded in the control loop, flight test were then documented to record the data for a range of commands at various initial conditions.

6.2.2 Lateral Response

Using the gains found in the previous section, the data recorded from the flight test will be analyzed. The sampling rate of the data-logging was being sampled at 25 *Hz* while the controller was operating at 5 *Hz*. This means for roughly every fifth data sample a control is send up to command the airplane. A control systems for an aircraft commanding on 5 *Hz* is unreasonable to expect a high level of performance. These sampling frequencies are an estimate and therefore every fifth data point cannot be plotted against the command. So to try and give an estimate of the response of the MAV during the command an average of the measured roll angle between control changes was used to plot the response, and is shown in Figure 6–3.

This figure shows the response of the MAV during a human commanded roll doublet. From this response plot, one can see reasonable transient response during the command from level to roughly $\pm 18 deg$. A steady state offset is shown for the first bank command and for the level command, which can be adjusted by slightly increasing the integral. For both bank commands of $\pm 18 deg$, there exists some oscillations which could be caused from several different factors. This oscillation could be a dutch roll mode being excited, a resolution issue, or a response to wind gust corrections. If these oscillations are caused by the dutch roll mode, a yaw damper can be used to control this mode.

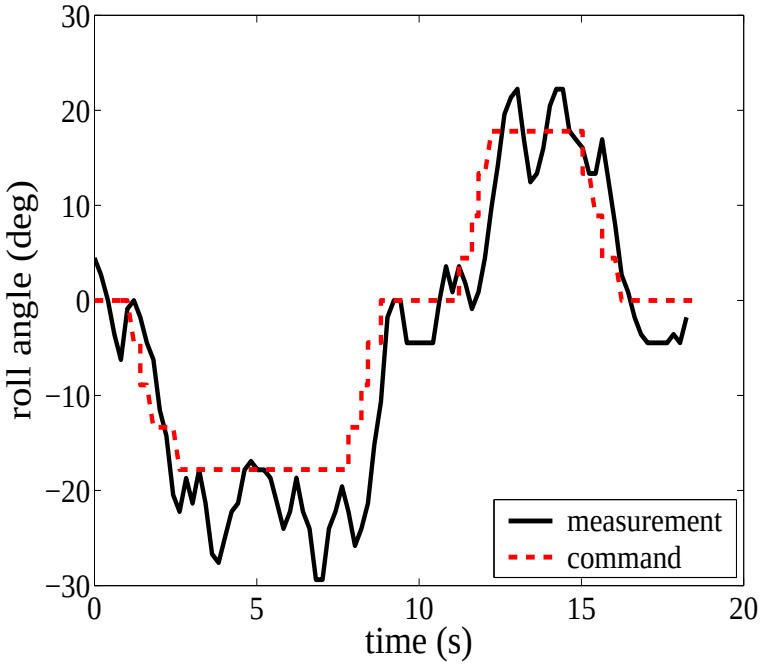


Figure 6–3: Roll Response

6.3 Future Work

There was some difficulty testing the heading controller and determining K_ψ. The GPS had frequent dropouts at the test site and even had difficultly locking on to satellites to acquire data. Some simple alterations can be made, such as placing copper tape around the sensor to increase the antenna area to acquire a stronger signal. The GPS will then be used for waypoint tracking, where longitude and latitude are used as feedback for the outer loop to control both heading and roll inner loops.

The current roll controller, described in this thesis, will require some alterations to reduce the oscillation observed during steady state, such as implementing a yaw damper. The yaw damper can also aid in stabilizing the aircraft during small wind gust as well. The longitudinal control system needs to be designed and implemented for an altitude hold where the inner loop commands a climb rate. Combining these two controller system, GPS waypoint tracking would be ready for implementation.

The next steps for vision-based autonomy would include the fusion of vision sensors with the standard inertia sensors. This could be used for increasing the

sampling rate, sensor bias corrections, and state estimation. The modeling of MAVs will be conducted through flight data acquired from testing and wind tunnel data in the lab. In the future, the models will aid in the control design process through simulation.

CHAPTER 7
CONCLUSION

In conclusion, this thesis has demonstrated that Micro Air Vehicles (MAVs) are a reasonable platform for vision-based autonomous research. The military describes many scenarios where this technology can be applied to save lives and help protect our country from possible terrorist attacks.

A final relation was derived between the dynamics of a MAV and the motion of feature points viewed by a camera attached to this vehicle. These equations, being nonlinear, then gave some insight into how objects are viewed as the MAV passes around them. This revealed how depth could change the shape of an object when its mapped into the focal plane. Another nonlinearity related to the camera was described as lens curvature. This mapping effected the outside portion of the image by warping feature points into a convex and concave pattern.

A physical feature was then extracted from the images to control the MAV, where the horizon line was determined. The image processing used a linear fit to track the horizon line. Using the horizon also allowed the camera nonlinearities to be neglected in the control design therefore, making the horizon's roll angle an estimate of the aircraft's roll angle.

Controller designs for the lateral states using roll and heading angle feedback were described using a classical proportional/integral control approach. With the design and fabrication of a 24 *in* MAV, a test-bed was constructed along with the required hardware and software to test the autopilot controls system. Due to some GPS dropouts, data was only taken for the roll controller. A response to a roll doublet was plotted and verified several issues. First, some steady state offset occurs during a bank hold with some oscillations possibly due to a dutch roll mode. Second, the

synchronization between data-logging and the controller was inconsistent with the data sampling a 25 *Hz* and the control working at 5*Hz*. To get an approximate response of the MAV during the doublet, an average measured roll angle was taken over the course of a control pulse.

The roll response acquired from the data, was tracking with a reasonable performance for a 5 *Hz* controller. For future works, the data-logger and the control should be operating at the same frequency to avoid any uncertainty in how the MAV is responding. Once this issue is resolved, flight test for the heading controller will preclude to determine the proportional gain. Finally, implementing a longitudinal controller to track altitude to complete the aircraft states. Further research will then lead onto implementing these controls systems into a fully autonomous waypoint tracker.

REFERENCES

[1] Akec, J.A., Steiner, S.J., and Stenger, F., " An Experimental Visual Feedback Control System for Tracking Applications Using a Robotic Manipulator," *IEEE*, 0-7803-4503-7/98, pp. 1125-1130.

[2] Bonnin, P., Stasse, O., Hugel, V., Blazevic, P., M'Sirdi, N., Coiffet, P., "How to Extract and to Exploit Vision Data for Autonomous and Mobile Robots to Operate in Known Environments," *IEEE International Workshop on Robot and Human Interactive Communication*, 2001, pp. 231-236.

[3] Bosse, M., Karl, W.C., Castanon, D., and DeBitetto, P., "A Vision Augmented Navigation System," *IEEE*, 0-7803-4269-0/97, 1998, pp. 1028-1033.

[4] Castro, A.P.A., Silva J. D., Simoni, P.O., "Image Based Autonomoue Navigation with Fuzzy Logic Control," *IEEE*, 0-7803-7044-9/01, 2001, pp. 2200-2205.

[5] Chatterji, G.B., "Vision-Based Position and Attitude Determination for Aircraft Night Landing," *AIAA Guidance, Navigation and Control Conference*, AIAA-96-3821, July 29-31, 1996.

[6] Choi, J.Y., Kim, C.S., Hong, S., Lee, M.H., Bae, J. II, and Harashima, F., "Vision Based Lateral Control by Yaw Rate Feedback," *The 27th Annual Conference of the IEEE Industrial Electronics Society*, 2001, pp. 2135-2138.

[7] Cohen, R., Hunt, T., and Seeliger, O., "Vision Based Control and Navigation of an Underwater Telerbot," *AIAA, Guidance, Navigation and Control Conference*, AIAA-96-3899, July 29-31, 1996.

[8] Ettinger, S.M., Nechyba, M.C., Ifju, P.G., and Waszak, M.R., "Vision-Guided Flight Stability and Control for Micro Air Vehicles," *IEEE International Conference on Intelligent Robots and Systems*, October 2002.

[9] Fleischer, S.D., Rock, S.M., and Burton, R., "Global Position Determination and Vehicle Path Estimation from a Vision Sensor for Real-Time Video Mosaicking and Navigation," *IEEE*, 0-7803-4108-2/97, 1997, pp.641-647.

[10] Franke, U., and Heinrich, S., "Fast Obsticle Detection for Urban Traffic Situation," *IEEE Transactions on Intelligent Transporation Systems*, Vol. 3, No. 3, September 2002, pp. 173-181.

[11] Gandhi, T., Yang, M.T., Kasturi, R., Camps, O., Coraor, L., and McCandless, J., "Detection of Obstacles in the Flight Path of an Aircraft," *IEEE Transaction*, 1063-6919/00, 2000.

[12] Garcia, H., Abdulrahim, M., and Lind, R., “Roll Control for a Micro Air Vehicle Using Active Wing Morphing,” *AIAA Guidance, Navigation and Control Conference*, Austin, TX, AIAA-2003-5347, August 2003.

[13] Grasmeyer, J.M., and Keennon, M.T., “Development of the Black Widow Micro-Air Vehicle,” AIAA 2001-0127, 2001.

[14] Gurfil, P., and Rotstein, H., “Partial Aircarft State Estimation from Visual Motion Using the Subspace Contraints Approach,” *Journal of Guidance, Control, and Dynamics*, Vol. 24, No. 5, September-October 2001, pp. 1016-1028.

[15] Herve, J.Y., “ Visual Feedback for Autonomous Navigation,” *IEEE*, 0-7803-0233-8/91, 1991, pp. 219-224.

[16] Huster, A., Fleischer, S.D., and Rock, S.M., “Demonstration of a zvision-Based Dead-Reckoning System for Navigation of an Underwater Vehicle,” *IEEE*, 0-7803-5045-6/98, 1998, pp. 326-330.

[17] Ifju, P.G., Jenkins, D.A, Ettinger, S., Lian, Y., Shyy, W., and Waszak, M.R., “Flexible-Wing-Based Micro Air Vehicles,” AIAA 2002-0705, 2002.

[18] Jung, J.S., and Tomlin, C.J., “Autopilot Design for the Stanford DragonFly UAV: Validation through Hardware-in-the-Loop Simulation,” *AIAA Guidance, Navigation and Control Conference and Exhibt*, AIAA 2001-4179, 2001.

[19] Kasetkasem, T., and Varshney, P.K., “An Image Detection Algorithm Based on Markov Random Fields Models,” *IEEE Transactions on Geoscience and Remote Sensing*, Vol. 40, No. 8, August 2002, pp. 1815-1823.

[20] Kato, T., Ninomiya, Y., and Masaki, I., “An Obsticle Detection Method by Fusion of Radar and Motion Stereo,” *IEEE Transactions on Intelligent Transportation Systems*, Vol. 3, No. 3, September 2002, pp. 182-188.

[21] Kimmett, J., Valasek, J., and Junkins J.K., “Autonomous Aerial Refueling Utilizing a Vision Based Navigation System,” *AIAA Guidance, Navigation and Control Conference*, AIAA-2002-4468, August 5-8, 2002.

[22] Lian, Y., and Shyy, W., “Three-Dimensional Fluid-Structure Interactions of a Membrane Wing for Micro Air Vehicle Applications,” *44th AIAA/ASME/ASCE/AHS Structures, Structural Dynamics, and Materials Conference*, AIAA 2003 1726, April 2003.

[23] Kosecka, J., Blasi, R., Taylor C.J., and Malik, J., “Vision-Based Lateral Control of Vehicles,” *IEEE*, 0-7803-4269-0/97, 1998, pp. 900-905.

[24] Kosecka, J., Blasi, R., Taylor C.J., and Malik, J., “A Comparative Study of Vision-Based Control Strategies for Autonomous Highway Driving,” *Proceedings of the 1998 IEEE International Conference on Robotics & Automation*, May 1998, pp. 1903-1908.

[25] Pal, N.R., and Bezdek, J.C., “Complexity Reduction for “Large Image” Processing,” *IEEE Transactions on Systems, Man, and Cybernetics - Part B: Cybernetics*, Vol. 32, No. 5, October 2002, pp. 598-611.

[26] Manigel, J., and Leonhard, W., “ Vehicle Control by Computer Vision,” *IEEE Transactions on Industrial Electronics*, Vol. 39, No. 3, June 1992, pp. 181-188.

[27] Mandelbaum, R., McDowell, L., Bogoni, L., and Hanson, M., “Real-Time Stereo Processing, Obstacle detection, and terrain estimation from Vehicle-Mounted stereo Cameras,” *IEEE*, 0-8186-8606-5/98, 1998.

[28] McLauchlan, P.F., and Malik, J., “ Vision for Longitudinal Vehicle Control,” *IEEE*, 0-7803-4269-0/97, 1997, pp. 918-923.

[29] Mojsilovic, A., Hu, J., and Soljanin, E., “Extraction of Perceptually Important Colors and Simiarity Measurement for Image Matching, Retrival, and Analysis,” *IEEE Tranactions on Image Processing*, Vol. 11, No. 11, pp. 1238-1248.

[30] Morris, D.D., and Kanade, T., “Image-Consistent Surface Triangulation,” *CVPR 2000*, Vol. 1, June 2000, pp. 332-338.

[31] Mueller, T.J. Editor, Proceedings of the Conference on Fixed, Flapping and Rotary Wing Vehicles at Very Low Reynolds Numbers, Notre Dame University, Indiana, June 5-7 2000.

[32] Raney, D.L., and Slominski, E.C., “Mechanization and Control Concepts for Bioloically Inspired Micro Air Vehicles,” *AIAA Guidance, Navigation, and Control Conference*, August 2003.

[33] Rock, S.M., Frew, E.W., Hank, J., LeMaster, E.A., and Woodley, B.R., “Combined CDGPS and Vision-Based Control of a Small Autonomous Helicopter,” *Proceedings of the American Control Conference*, 0-7803-4530-4/98, Jume, 1998, pp. 694-698.

[34] Sellers, W.L., III and Kjelgaard, S.O., “The Basic Aerodynamics Research Tunnel - A Facility Dedicated to Code Validation,” AIAA-88-1997, May, 1988.

[35] Silveira, G.F., Carvalho, J.R.H., Madirid, M.K., Rives, P., and Bueno, S.S., “ A Fast Vision-Based Road Following Strategy Applied to the Control of Aerial Robots,” *IEEE*, 1530-1834/01, 2001, pp. 226-231.

[36] Smith, J.S., Yu, R., and Lucas, J., “ Computer Vision Control of an Underwater Manipulator,” *IEEE*, 0-7803-2056-5, 1994, pp. I187-I192

[37] Soatto, S., Perona, P., Frezza, R., and Picci, G., “Motion Estimation via Dynamic Vision,” *IEEE Proceedings of the 33rd Conference on Decision and Control*, December 1994, pp. 3253-3258.

[38] Soatto, S., and Perona, P., “Dynamic Visual Motion Estimation from Subspace Constraints,” *IEEE*, 0-8186-6950-0/94, 1994, pp. 333-337.

[39] Soatto, S., and Perona, P., “Reducing ’Structure from Motion’,” *IEEE*, 1063-6919/96, 1996, pp. 825-832.

[40] Soatto, S., and Perona, P., “Visual Motion Estimation from Point Features: Unified View,” *IEEE*, 0-8186-7310-9/95, 1995, pp. 21-24.

[41] Soatto, S., and Perona, P., “Reducing ’Structure from Motion’: A General Framework for Dynamic Vision Part 1: Modeling,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 20, No. 9, September 1998, pp.933-942.

[42] Soatto, S., and Perona, P., “Reducing ’Structure from Motion’: A General Framework for Dynamic Vision Part 2: Implementation and Experimental Assessment,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 20, No. 9, September 1998, pp. 943-960.

[43] Sznaier, M., and Camps, O.I., “Control Issues in Active Vision: Open Problems and Some Answers,” *Proceedings of the 37th Conference on Decision & Control*, December 1998, pp. 3238-3244.

[44] Shyy, W., Berg, M., and Ljungqvist, D., “Flapping and Flexible Wings for Biological and Micro Air Vehicles,” *Progress in Aerospace Sciences*, Vol. 35, No. 5, 1999, pp. 455-506.

[45] Wasack, M.R., Davidson, J.B., and Ifju, P.G., “Simulation and Flight Control of an Aeroelastic Fixed Wing Micro Air Vehicle,” *AIAA Atmospheric Flight Mechanics Conference*, AIAA 2002-4875, 2002.

[46] Wasack, M.R., Jenkins, L.N., and Ifju, P.G., “Stability and Control Properties of an Aeroelastic Fixed Wing Mirco Air Vehicle,” *AIAA Atmospheric Flight Mechanics Conference*, AIAA 2001-4005, 2001.

[47] Valasek, J., Kimmett, J., Hughes, D., Gunnam, K., and Junkins, J.L., “Vision Based Sensor and Navigation for Autonomous Aerial Refueling,” *AIAA 1st Technical Conference and Workshop on Unmanned Aerospace Vehicles, Technologies, and Operation*, AIAA 2002-3441, May 20-23, 2002.

[48] Zang, B., Lian, Y., and Shyy,W., “Proper Orthogonal Decomposition for Three-Dimensional Membrane Wing Aerodynamics,”*33rd AIAA Fluid Dynamics Conference and Exhibit*, AIAA 2003-3917, June 2003.

[49] Zang, M., Hall, L.O., and Goldgof, D.B., “A Generic Knowledge-Guided Image Segmentation and Labeling System Using Fuzzy Clustering Algorithms,” *IEEE Transactions on Systems, Man, and Cybernetics - Part B: Cybernetics*, Vol. 32, No. 5, October 2002, pp. 571-582.

BIOGRAPHICAL SKETCH

Ryan Scott Causey was born in Miami, Florida, on May 10, 1978. He grew up in a stable family with one brother in a typical suburban home. During his teenage years and into early adolescence, Ryan built and maintained a small business providing lawn care to the local neighborhood. The tools acquired from this work carried over into his college career. After graduating from Miami Killian Senior High School, Ryan attended Miami Dade Community College for two years that commenced with an Associate in Arts degree in engineering. A transfer student to the University of Florida, Ryan was prepared to tackle the stresses of a university aside from the poor statistics. A few years later, he received a Bachelor of Science in Aerospace Engineering degree with honors and was considered in the top three of his class. During his summers before and after graduation, he worked for Honeywell Space Systems in Clearwater, Florida, as an intern applying his education to guided defense missiles. Ryan soon after chose to attend graduate school back at the University of Florida under Dr. Andrew Kurdila and Dr. Richard Lind in the Dynamics and Controls Laboratory. Vision-based control of air-vehicles became his interest and he is now pursuing a doctorate degree on this topic.