2005-07-07

Vision-based Target Localization from a Small, Fixed-wing Unmanned Air Vehicle

Joshua D. Redding
Brigham Young University - Provo

VISION-BASED TARGET LOCALIZATION FROM A SMALL, FIXED-WING UNMANNED AIR VEHICLE

by

Joshua David Redding

A thesis submitted to the faculty of

Brigham Young University

in partial fulfillment of the requirements for the degree of

Master of Science

Department of Mechanical Engineering

Brigham Young University

August 2005

BRIGHAM YOUNG UNIVERSITY

GRADUATE COMMITTEE APPROVAL

of a thesis submitted by

Joshua David Redding

This thesis has been read by each member of the following graduate committee and by majority vote has been found to be satisfactory.

Date Timothy W. McLain, Chair

Date Randal W. Beard

Date W. Jerry Bowman

BRIGHAM YOUNG UNIVERSITY

As chair of the candidate's graduate committee, I have read the thesis of Joshua David Redding in its final form and have found that (1) its format, citations, and bibliographical style are consistent and acceptable and fulfill university and department style requirements; (2) its illustrative materials including figures, tables, and charts are in place; and (3) the final manuscript is satisfactory to the graduate committee and is ready for submission to the university library.

Date

Timothy W. McLain
Chair, Graduate Committee

Accepted for the Department

Matthew R. Jones
Graduate Coordinator

Accepted for the College

Alan R. Parkinson
Dean, Ira A. Fulton College of
Engineering and Technology

ABSTRACT

VISION-BASED TARGET LOCALIZATION FROM A SMALL, FIXED-WING UNMANNED AIR VEHICLE

Joshua David Redding

Department of Mechanical Engineering

Master of Science

Unmanned air vehicles (UAVs) are attracting increased attention as their envelope of suitable tasks expands to include activities such as perimeter tracking, search and rescue assistance, surveillance and reconnaissance. The simplified goal of many of these tasks is to image an object for tracking or information-gathering purposes. The ability to determine the inertial location of a visible, ground-based object without requiring a priori knowledge of its exact location would therefore prove beneficial.

This thesis discusses a method of localizing a ground-based object when imaged from a fixed-wing UAV. Using the target's pixel location in an image, with measurements of UAV position, attitude and camera pose angles, the target is localized in world coordinates. This thesis also presents a study of possible error sources and localization sensitivities to each source. From this study, an accuracy within 15.5 m of actual target location is expected. Also, several methods of filtering are presented, which allow for effective noise reduction. Finally, filtered hardware results are presented that verify these expectations by localizing a target from a fixed-wing UAV using on-board vision to within 10.9 meters.

ACKNOWLEDGMENTS

This thesis is the result of many contributions made by co-workers, friends and family. I would like to thank these contributors for their time, talents, and most of all for their patience. In particular, Reed Christiansen, Dave Hubbard, Derek Kingston, Brandon Call, Dave Johansen, Steve Griffiths and Andrew Eldredge. Also, Professors Tim McLain and Randy Beard for their insight and guidance from inception through implementation of this project. Last and certainly not least of all, my wife Erica for her constant support. Thank you.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Although unmanned vehicles have been in use since the 1970's, the last decade has seen a dramatic rise in both interest and application as the needed technology has continued to shrink in size, weight and cost while showing significant increases in capability. For example, the "Predator", General Atomics' 27 foot UAV, has logged over 20,000 flight hours since 1994, solidly proving its reliability and functionality [1]. Likewise, Northrop Grumman's "Global Hawk" with its 116 foot wingspan, 1900 pound payload capacity and 40,000 daily searchable square nautical miles has proved its effectiveness with over 50 completed combat missions under Operation Enduring Freedom in Afghanistan [1]. With track records such as these, it is no wonder that over the next five years more than 12 billion dollars will be spent bringing UAVs to the forefront of military and military support operations [2].

1.1 Motivation

The nature of tasks generally suited for UAVs motivates their further development. Unmanned systems are prime candidates for activities involving risk or repetition, what the military calls the "dull, dirty and dangerous" [2]. UAVs represent a technology that is capable of saving countless lives by performing dangerous tactical and reconnaissance missions, decreasing response time in bringing aid to victims of natural disasters and decreasing search times in search and rescue missions. The simplified goal of many of these tasks is to locate and image an object for tracking, information-gathering or delivery purposes. The ability to determine the inertial location of a visible, ground-based object without requiring a priori knowledge of its precise location is beneficial to the completion of these tasks. UAVs show a strong potential for accommodating such a capability. Many of the current approaches to this object localization problem involve imaging and localizing a target from unmanned blimps or rotor craft [3, 4, 5]. Due to their low-velocity and low-altitude flight capabilities, these aircraft allow the target to occupy a larger percentage of an image, making vision-based localization much easier. However, blimps are not suited for use in wind or weather, and the cost and complexity associated with rotor craft are high. It is therefore reasonable to explore target localization methods involving more robust and less-expensive flight platforms.

Fixed wing UAVs, while lacking the ability to hover, present unique benefits such as adaptability to adverse weather, a shorter learning curve for the untrained operator and extreme durability against harsh environments. Also, the multiple vantage points offered by a fixed-wing UAV provide more usable information for localization, tracking and other purposes. Man-packable, fixed wing UAVs are small enough to be carried and operated by a single person while offering many of the same advantages as their larger counterparts, which explains their recent increase in demand [6]. The development and deployment of such UAVs is relatively inexpensive, making them comparatively expendable [7]. This presents a significant advantage over other alternatives since expendable units can be justifiably placed in even the most dangerous situations without incurring unnecessary risk. Also, many of these small UAVs are still large enough to carry an on-board camera. An aerial camera can efficiently relay large amounts of information to the user/operator. In addition, this research will use the video stream as a means of localizing objects found in its field of view. Vision-based object localization represents an effective use of on-board resources and provides information for many practical applications.

1.2 Problem Description

The problem exists in many applications that a target needs to be found or photographed when its exact location is not known. Such is often the case in both military and search and rescue scenarios. The goal of this research is to determine the location of an arbitrary ground-object in world coordinates using a gimbaled camera on-board a fixed-wing UAV. The general, rather than exact, location of the target is assumed to be known a priori. The process of estimating the target's exact location is called localization and provides similar information about the target as the global positioning system (GPS). Since all images lack a depth dimension, vision-based localization is a non-trivial problem. The image depth, also called λ, is a measure of the distance from the camera to the object of interest in the image. This value is often unknown and must therefore be estimated for every snapshot taken of the target. This research presents a simple method for estimating λ from UAV altitude measurements. The result is then used to derive the exact location of the ground-based target using noisy and imperfect sensor data that includes UAV location and attitude as well as camera pose angles. This method generates a location estimate to accompany each aerial photo taken of the target, which occurs frequently enough to allow for effective noise-reducing methods. In order to discover reasonable accuracy expectations, a study of possible error sources and their propagation into the final estimate of target location is also conducted.

The crux of this research is determining the inertial, or world location of a target from visual data. However, this objective presents several smaller problems such as initially finding and recognizing the target. A solution was found by allowing the UAV to follow a prescribed search pattern that covers the target's known general area while a color-segmentation routine picks out the highly-visible target from its surroundings when it enters the camera's field of view. The result of the color-segmentation routine is the (x, y) pixel location of the target in the camera image. This pixel location corresponds to the object's inertial location when λ is known.

1.3 Previous Work

Current research in localization generally involves an application to mobile-robots. The main focus areas are GPS/predictive-based and vision-based localization, both of which have some degree of relevancy to this research and will be discussed accordingly.

1.3.1 GPS/Predictive Solutions

In attempting to localize a ground-based object from a UAV, it is important to have accurate information for aircraft attitude and position. For this research, a simple first-order prediction is used to estimate UAV location and attitude in between GPS updates. Other research, particularly relating to ground vehicles, uses odometry to estimate robot position and orientation relative to an arbitrary origin. Technology has greatly aided these processes with devices such as accelerometers, rate gyros and optic flow sensors. Although relatively inexpensive, most of these sensors are insufficient when used alone. Barshan [8] explains that small errors in gyros and accelerometers can result in unbounded error growth when their measurements are integrated to provide usable information; such as robot position, orientation, and/or velocity. Fitting an error model to the gyros can reduce the error growth to less than half a degree per minute. However, this is still unacceptable over time. This problem motivates the need for a supplemental reference to true position and orientation information. For this research, reference to GPS information flushes accumulated errors in estimated UAV heading and location.

1.3.2 Vision Based Solutions

Vision-based localization has been studied for many years, resulting in methods from optical flow to stereo and Monte Carlo [4, 9, 10, 11, 12]. Others, such as Marques [13], fuse odometry and computer vision by resetting predictive errors using updates from visual information of known landmarks. Although Marques uses an omni-directional vision system on-board a soccer robot, his method of localization represents a similar technology desired for this research. The main difference being that we will estimate the landmark's position when robot (UAV) position is known rather than the converse. Also, this research will extend these vision-based techniques from structured, indoor environments to unstructured, outdoor environments.

Rysdyk [14] conducted simulations to maintain constant line-of-sight with a ground-based target from a fixed-wing UAV. He outlines camera pan and tilt control, but emphasizes UAV path-planning for constant target viewing and assumes a priori knowledge of target location. Stolle [15] presents similar research, but with more details on camera control. Both Rysdyk and Stolle assume a camera center located at the UAV center of mass. This thesis extends the simulated camera control done by Rysdyk and Stolle into hardware implementation and accounts for added complexities such as an arbitrary camera location.

Aside from camera control, this research most closely relates to that of Chaimowicz [3], who presented the results of an experiment using a blimp to localize a stationary object on the ground. Flying over the target at an altitude of 18 m and equipped with a fixed camera, two methods of target localization were employed and compared with the target's actual GPS location. First, a series of images were taken where both the target and a set of at least 6 known landmarks were visible. Although this method led to very accurate localization, it was deemed unreliable since acquiring a sufficient number of landmarks in the image is not guaranteed. Second, GPS and IMU measurements from the blimp were used to localize the target, which they achieved within 8.2 m. This thesis will employ a similar method of combining GPS and IMU information to localize a target. In addition, this experiment will include a gimbaled camera mounted to a fixed-wing UAV flying at a nearly constant velocity of 13 m/s and at a much higher altitude of 60 m. These additions introduce new depth and practicality to the vision-based target localization problem.

1.4 Contributions

This thesis will discuss the development and hardware testing of a robust algorithm for a fixed-wing UAV to localize a computer-recognizable ground object. As discussed in the previous section, vision-based localization is not a new concept. Most previous experiments, however, have used similar vision-based techniques for unmanned ground vehicle self-localization, either against other robots or landmarks in an unknown environment. The contributions of this research extend the application of vision-based localization methods to fixed-wing UAVs. This, combined with an inexpensive UAV platform, represent a very practical technology.

1.5 Outline

This thesis is organized as follows: Chapter 2 gives background that explains the groundwork for this experiment. The localization method is detailed in Chapter 3. Chapter 4 presents a study of error sources and their propagation effects. Hardware test results are presented in Chapter 5 with conclusions and recommendations for future work found in Chapter 6.

Chapter 2

Background

The purpose of this chapter is to provide enough background information to give the reader an accurate idea of the foundation on which this project was built and the tools that helped take it to hardware demonstration. The functionality and purpose of these tools will be discussed. An overview of the functional specifications and capabilities of the UAV autopilot is outlined in section 2.1. A description of the fixed-wing UAV platform follows in section 2.2. Section 2.3 describes the software developed at BYU to control the UAV from the ground. Section 2.4 briefly explains the vision routine used to locate the object in the camera's image. The problem is then formally defined in section 2.5

2.1 UAV Autopilot

The Kestrel AutoPilot (KAP) was developed at Brigham Young University to provide a solution for the intelligent and autonomous flight control of micro/mini air vehicles. It is comprised of a fully integrated processor and sensor suite and is commercially available through Procerus Technologies [16].

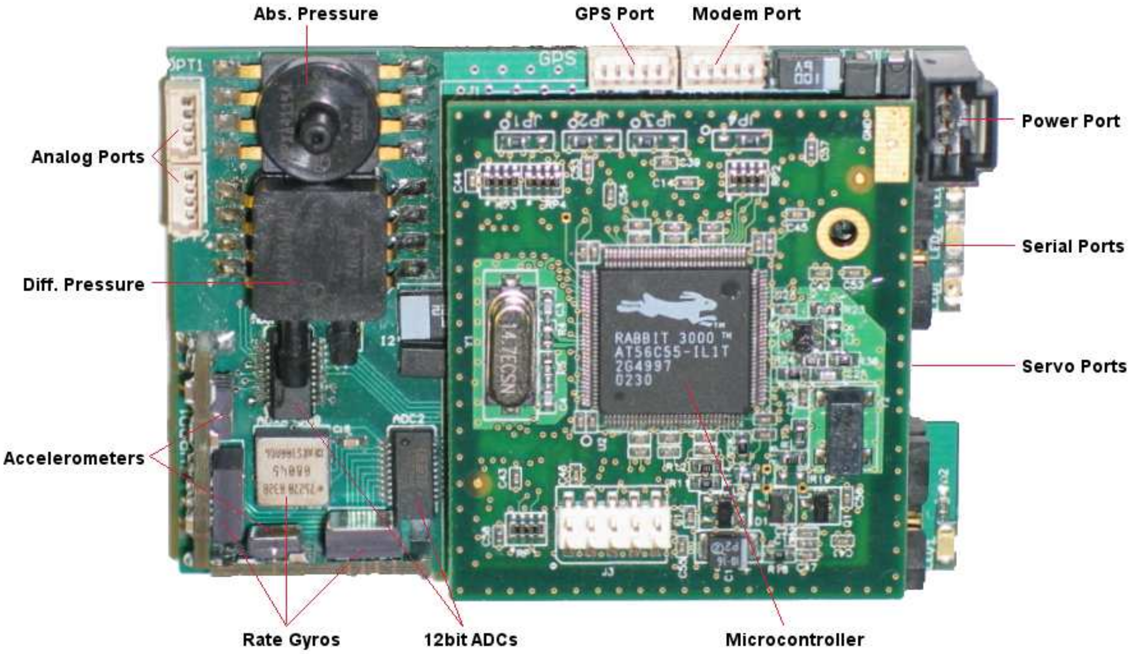
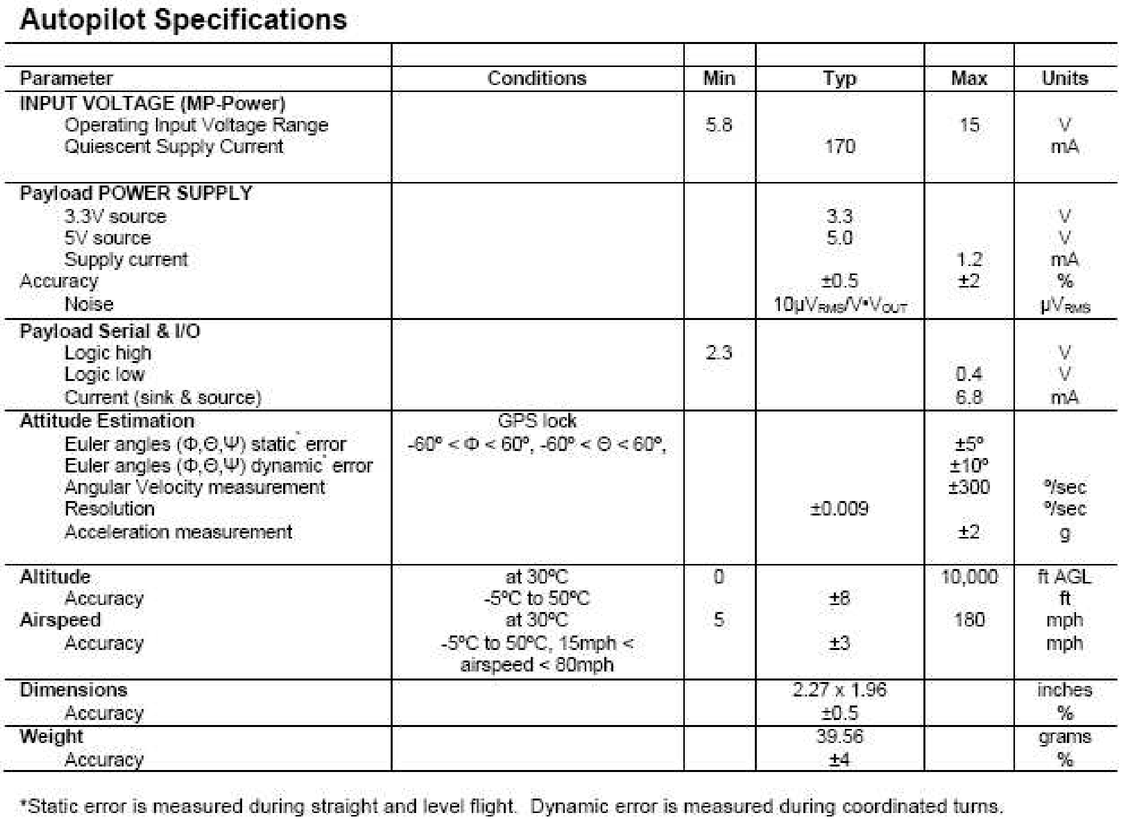


Figure 2.1: Kestrel AutoPilot (KAP) 1.45 and components

Figure 2.1 shows the autopilot and Figure 2.2 [16] gives some key technical specifications for the KAP 1.45, which was used for this experiment. The KAP 1.45 is equipped with a sensor suite including two dual-axis accelerometers, three single-axis rate gyros, absolute and differential pressure sensors and four additional analog input pins for payload needs. Using these sensors, combined with GPS information, the KAP estimates its own attitude, inertial position between GPS updates, altitude and airspeed. Knowing these parameters allows the KAP to facilitate autonomous take-offs, landings and GPS waypoint navigation. Its small size and extreme light weight allow UAVs as small as 12 inches to perform these tasks with a single user [16]. For a thorough and technical explanation of the KAP, including each of its components, see [6].



Autopilot Specifications

Parameter	Conditions	Min	Typ	Max	Units
INPUT VOLTAGE (MP-Power)					
Operating Input Voltage Range		5.8		15	V
Quiescent Supply Current			170		mA
Payload POWER SUPPLY					
3.3V source			3.3		V
5V source			5.0		V
Supply current				1.2	mA
Accuracy			±0.5	±2	%
Noise			$10\mu V_{RMS}/V \cdot V_{OUT}$		μV_{RMS}
Payload Serial & I/O					
Logic high		2.3			V
Logic low				0.4	V
Current (sink & source)				6.8	mA
Attitude Estimation	GPS lock				
Euler angles (Φ,Θ,Ψ) static* error	-60° < Φ < 60°, -60° < Θ < 60°,			±5°	
Euler angles (Φ,Θ,Ψ) dynamic* error				±10°	
Angular Velocity measurement				±300	°/sec
Resolution			±0.009		°/sec
Acceleration measurement				±2	g
Altitude	at 30°C	0		10,000	ft AGL
Accuracy	-5°C to 50°C		±8		ft
Airspeed	at 30°C	5		180	mph
Accuracy	-5°C to 50°C, 15mph < airspeed < 80mph		±3		mph
Dimensions			2.27 x 1.96		inches
Accuracy			±0.5		%
Weight			39.56		grams
Accuracy			±4		%

*Static error is measured during straight and level flight. Dynamic error is measured during coordinated turns.

Figure 2.2: Kestrel AutoPilot (KAP) 1.45 specifications

2.2 UAV Platform

A flight platform consists of an airframe and all of the necessary hardware to keep it aloft. Specifically, this includes the control-surface actuators, electronic speed control, power and propulsion system. Christiansen [6] indicates that a good UAV flight platform should be small enough to be portable, capable of carrying the autopilot and payload, resistant to damage caused by inevitable abuse and be able to stay airborne long enough to complete a useful mission. With these guidelines in mind, a commercially available Zagi^{TM} airframe was chosen for this and virtually all other UAV experiments performed at BYU.

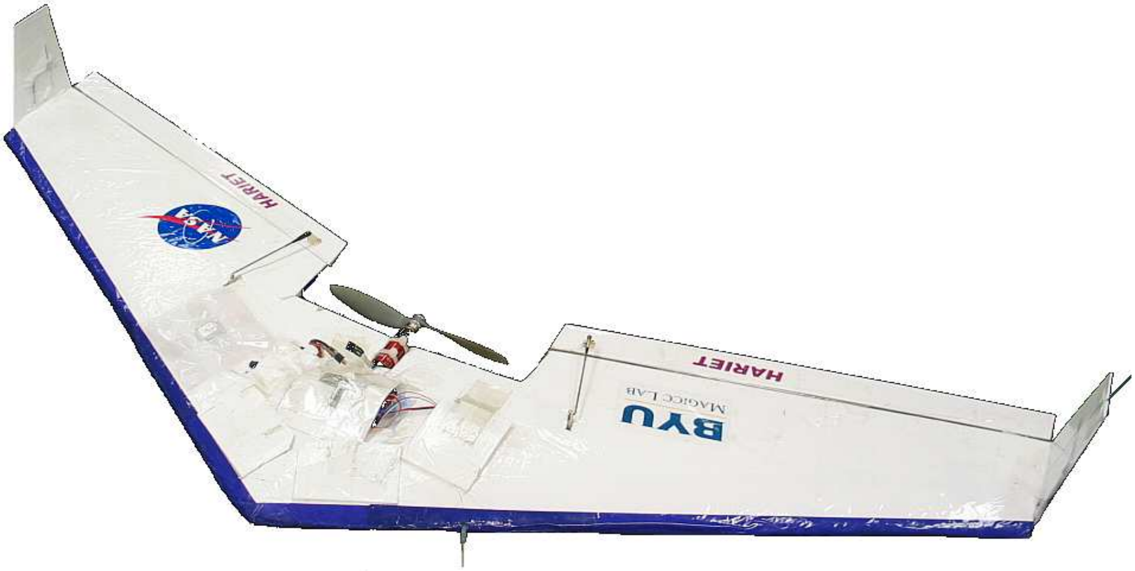


Figure 2.3: Finished UAV: Autopilot, Gimbal, GPS etc..

The Zagi THL is a simple flying wing design constructed of extremely durable, yet pliable EPP foam, which can aptly tolerate field abuses. The foam wing design provides plenty of volume for embedding avionics, antennas and payload components. Figure 2.3 shows a finished Zagi airframe ready for autonomous flight after modifications and upgrades. In addition to all other platform requirements, the UAVs used for this experiment carry a two-axis gimbaled camera, shown in Figure 2.4. Gimbal azimuth and elevation angles are controlled indirectly by the KAP through a separate servo-controller board.

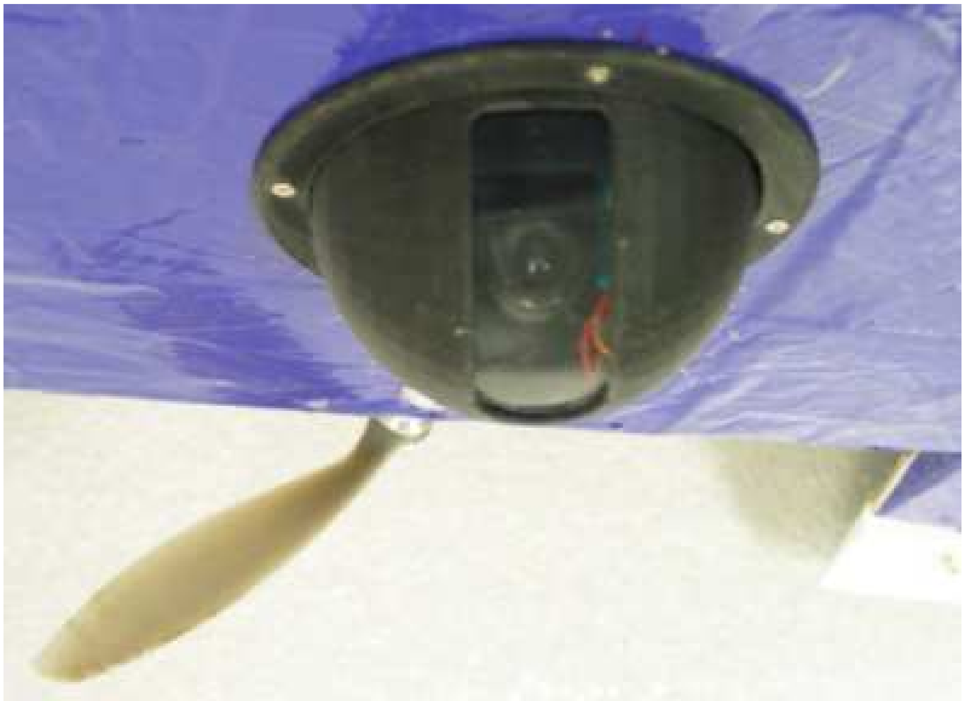
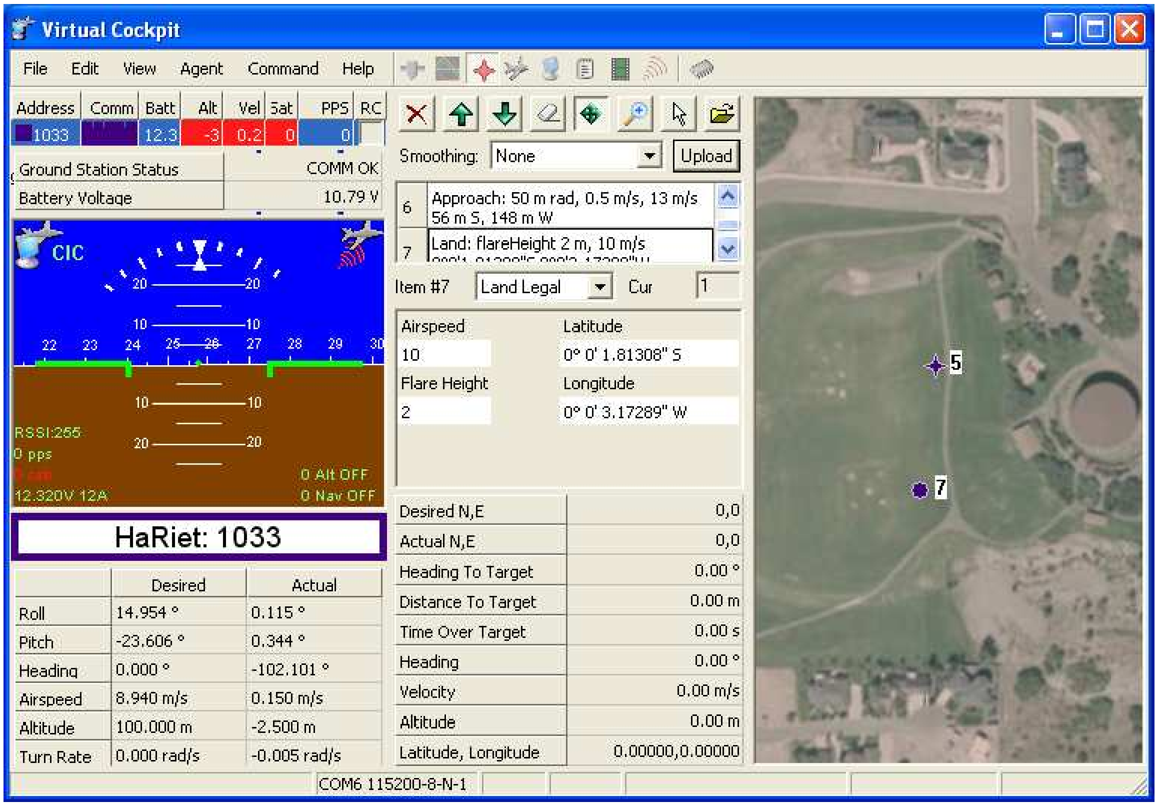


Figure 2.4: Two-axis Gimbal

2.3 Virtual Cockpit / Ground Station

The UAV is operated via a software program running on a laptop computer, which is supervised by an operator. The software, appropriately named the Virtual Cockpit, is shown in Figure 2.5 and was developed at Brigham Young University. The Virtual Cockpit is simply an interface for the flight control of UAVs. It is capable of on-the-fly gain tuning, data and telemetry logging, displaying streamed video in real-time, uploading and modifying GPS waypoints and displaying the UAV's attitude and location. The Virtual Cockpit communicates with the UAV via a custom communication box that relays autonomous flight control commands to the UAV and pilot commands from a Radio-Controlled transmitter. The KAP constantly interprets the information obtained its sensors and adjusts its actuators according to the flight plan uploaded from the Virtual Cockpit. The gimbaled camera receives azimuth and elevation commands from the KAP to keep the target locked in the center of the



Address	Comm	Batt	Alt	Vel	Sat	PPS	RC
1033		12.3	-3	0.2	0	0	

Ground Station Status	COMM OK
Battery Voltage	10.79 V

	Desired	Actual
Roll	14.954 °	0.115 °
Pitch	-23.606 °	0.344 °
Heading	0.000 °	-102.101 °
Airspeed	8.940 m/s	0.150 m/s
Altitude	100.000 m	-2.500 m
Turn Rate	0.000 rad/s	-0.005 rad/s

Smoothing: None

6 Approach: 50 m rad, 0.5 m/s, 13 m/s 56 m S, 148 m W

7 Land: flareHeight 2 m, 10 m/s

Item #7 Land Legal Cur 1

Airspeed	Latitude
10	0° 0' 1.81308" S
Flare Height	**Longitude**
2	0° 0' 3.17289" W

Desired N,E	0,0
Actual N,E	0,0
Heading To Target	0.00 °
Distance To Target	0.00 m
Time Over Target	0.00 s
Heading	0.00 °
Velocity	0.00 m/s
Altitude	0.00 m
Latitude, Longitude	0.00000,0.00000

Figure 2.5: Virtual Cockpit

image, but it can also be manually controlled via the Virtual Cockpit. The video from the gimbaled camera is piped down to the ground via a 2.4GHz transmitter/receiver and into the laptop via a framegrabber. This allows for real-time video display in the Virtual Cockpit and data extraction for use in tracking algorithms. The overall system architecture is mapped out in Figure 2.6.

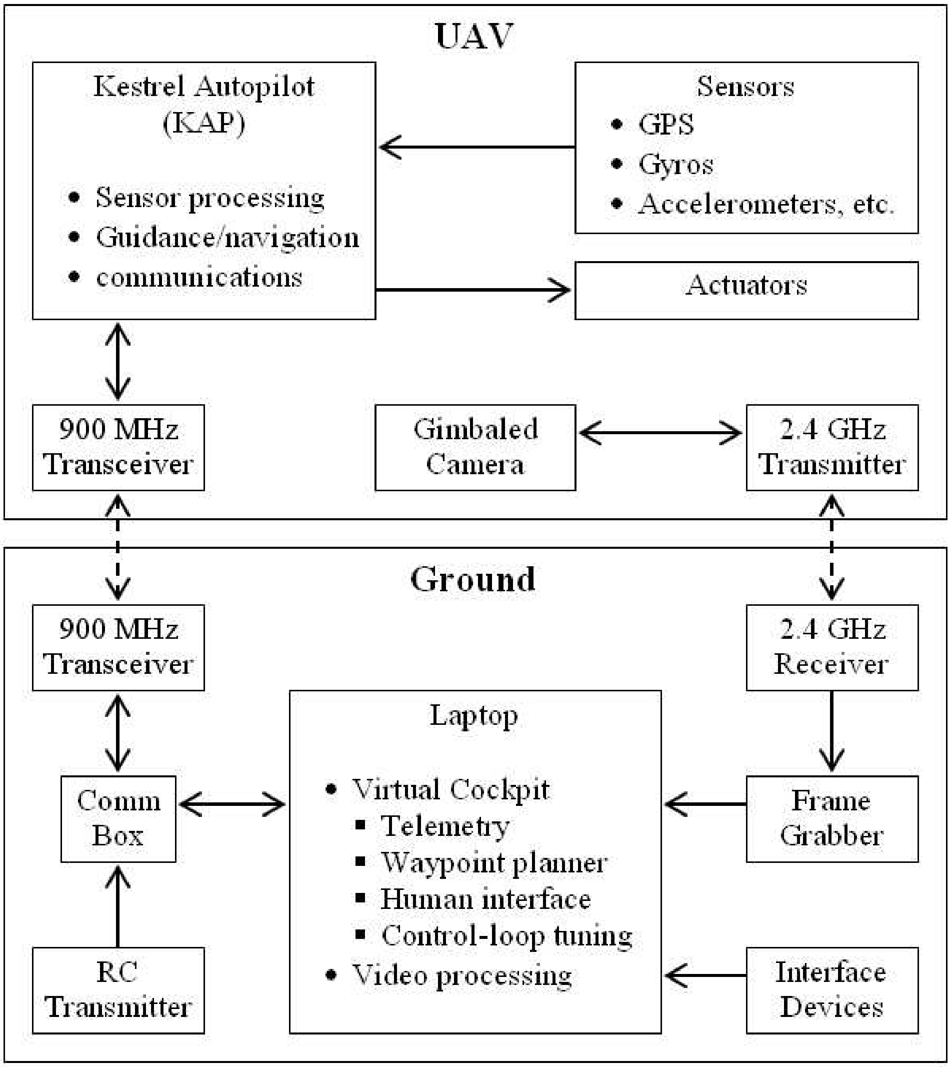
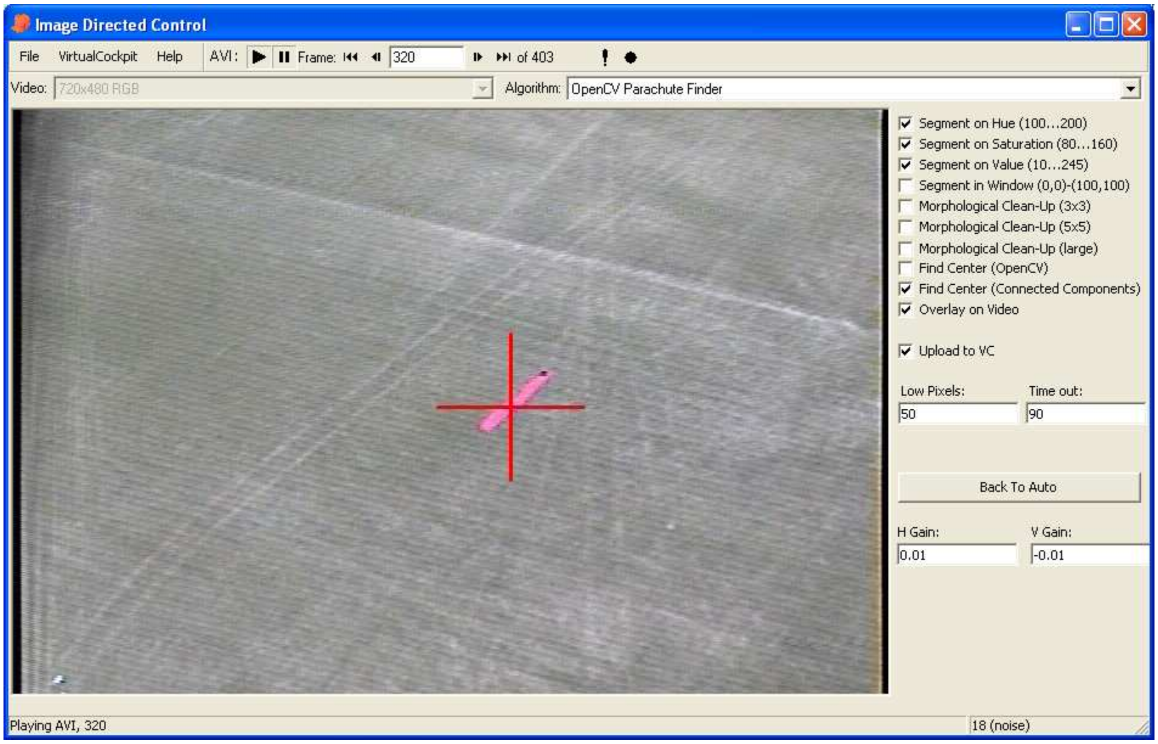


Figure 2.6: Setup of system architecture

2.4 Image Processing

A desirable goal is to track an object using only a camera. To do this, the tracked object must be distinctly visible in the camera's image. "Distinctly visible", however, is a relative phrase since what may seem so to the human eye, may or may not be to a computer vision algorithm. For a single sensor, a camera is able to communicate large amounts of information. A software program called Image Directed Control (IDC) was developed at Brigham Young University to track a target in a video stream and output its (x, y) pixel location in each frame. Figure 2.7 shows a screen grab of IDC and Figure 2.8 portrays the interaction between IDC and the experimental hardware, where "VC" represents the Virtual Cockpit and "SC" the servo controller board. IDC receives an image from the on-board camera, calculates a pixel location of the target in the image and sends this location to the Virtual Cockpit where it is used to calculate a world estimate of target location.



Frame: 320 of 403

Video: 720x480 RGB

Algorithm: OpenCV Parachute Finder

☑ Segment on Hue (100...200)
☑ Segment on Saturation (80...160)
☑ Segment on Value (10...245)
□ Segment in Window (0,0)-(100,100)
□ Morphological Clean-Up (3x3)
□ Morphological Clean-Up (5x5)
□ Morphological Clean-Up (large)
□ Find Center (OpenCV)
☑ Find Center (Connected Components)
☑ Overlay on Video

☑ Upload to VC

Low Pixels: 50

Time out: 90

H Gain: 0.01

V Gain: -0.01

Playing AVI, 320

18 (noise)

Figure 2.7: Image Directed Control (IDC)

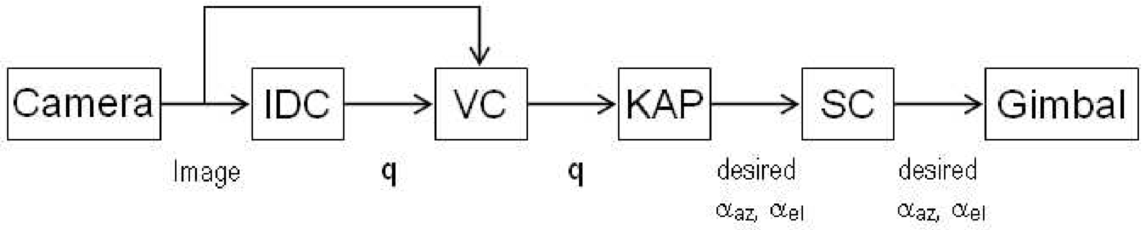


Figure 2.8: IDC/Hardware interaction

2.5 Problem Definition

A reasonable scenario for baseline testing of the problem stated in Section 1.2 is a single UAV localizing a highly visible target using only its gimbaled camera. For this experiment, a bright red cloth was used as the target and was placed in a known area, its GPS location known only for a truth reference. The UAV, launched from an arbitrary home position, then flew over the general area of the target while an operator monitored the live video from the UAV. When the operator saw the target the UAV was sent into a radial orbit over the user's guess of the target's location. While circling this location, the UAV's gimbal was actively controlled via IDC to keep the target centered in the video image. IDC calculated and sent target pixel locations to the Virtual Cockpit where the information was recorded and used to localize the target in the inertial frame. This newly estimated target location then updated the user's initial guess and the UAV adjusted its orbit center and gimbal angles accordingly. This procedure allowed the UAV to maintain an effective circular orbit directly over the approximated target location while keeping the target within the camera's field of view.

2.6 Summary

This project was built on a foundation of previous research conducted in the areas of computer vision and UAV control. The building blocks of this experiment include: 1) a functional UAV equipped with autopilot and gimbaled camera, 2) a software interface for communication with and control of the UAV, and 3) a color-segmentation algorithm inside a color-tracking routine that is capable of recognizing a target in its surroundings and producing the necessary commands to track it with a camera. Granted, minor adjustments and additions were necessary incorporate these building blocks into this experiment, however, without the foundation they provide, this research would not be possible. Now that these tools are in place, the next step is to introduce the mathematics of localization.

Chapter 3

Mathematics of Localization

The crux of this project is to determine the inertial location of a visible and stationary object from the vantage point of a fixed-wing UAV. The mathematics underlying such a task require measurements of the UAV's attitude and inertial position as well as its camera pose, i.e. azimuth and elevation angles. In addition, the localization algorithm must be robust to typical noise and disturbances in each of these measurements, since the resulting inaccuracies will be magnified by the UAV's altitude. An investigation of error sources and methods of error minimization is covered in Chapter 4.

In this chapter, an outline and derivation of the mathematical underpinnings of vision-based localization are presented. Using the target location in pixels from a UAV image, an estimate of the target location relative to the world reference frame is calculated. This estimate is continuously updated as more pictures of the target are captured, and sent to the UAV as the new desired center of its circular orbit path over the target. Section 3.1 begins by reviewing necessary terminology and outlining the process as a whole. Section 3.2 explains the steps of transformation between world and camera frame coordinates. Section 3.3 details the camera's calibration matrix, which correlates the target's three-dimensional camera frame coordinates with its two-dimensional pixel coordinates and a scale factor λ. A method for estimating this scale factor is discussed in Section 3.4. In conclusion, Section 3.6 provides a concise summary of the localization algorithm.

3.1 Getting Started

The purpose of this section is to discuss some important terms and concepts before delving into precise details. First, Section 3.1.1 shows and explains the sets of coordinate frames used in the localization routine. Second, the problem and proposed solution are thoroughly outlined in Section 3.1.2.

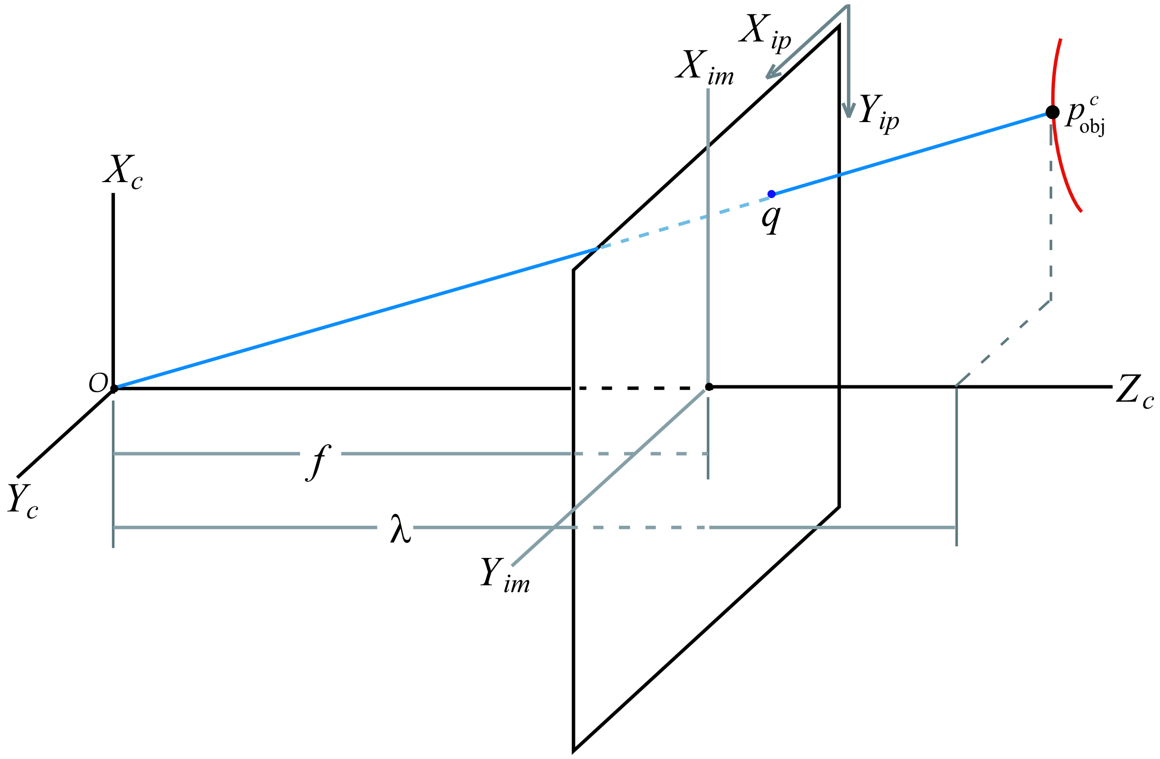


Figure 3.1: Camera frames

3.1.1 Coordinate Frames

Understanding Figures 3.1, 3.2, and 3.3 is critical to comprehending the following problem description and derivation. Figure 3.1 shows a target at location (x_c, y_c, z_c) relative to several coordinate frames, including the camera frame, denoted with subscript c, the image pixel frame, ip and the image meter frame, im. Figure 3.1

assumes a simple projection camera model and the origin is marked by O. The camera's focal length is denoted by f and has units of meters. However, this will later be scaled in the x and y directions and become f_x and f_y respectively, yielding units of pixels.

The individual coordinate frames associated with the UAV are shown in Figures 3.2 and 3.3, with subscripts defined in Table 3.1. All coordinate frames follow a right-hand rule. The origin of the camera frame is at the camera's center, with the positive camera Z-axis, Z_c, representing the optical axis of the camera. The origin of the gimbal frame is the center of the two-axis gimbal. The axes correspond to the UAV body axes when gimbal azimuth and elevation angles are zero. The UAV body frame is centered at the center of mass, with the X-axis, X_b, out the nose of the aircraft and the Y-axis, Y_b, out the right wing. The UAV vehicle frame is identical to the inertial frame but is translated so the UAV center of gravity is the origin. It will prove helpful for the reader to become familiar with these coordinate frames.

Table 3.1: Index to coordinate frame sub/superscripts.

Coordinate Frame	subscript
Image frame (pixels)	ip
Image frame (meters)	im
Camera frame	c
Gimbal frame	g
UAV Body frame	b
UAV Vehicle frame	v
World/Inertial frame	I

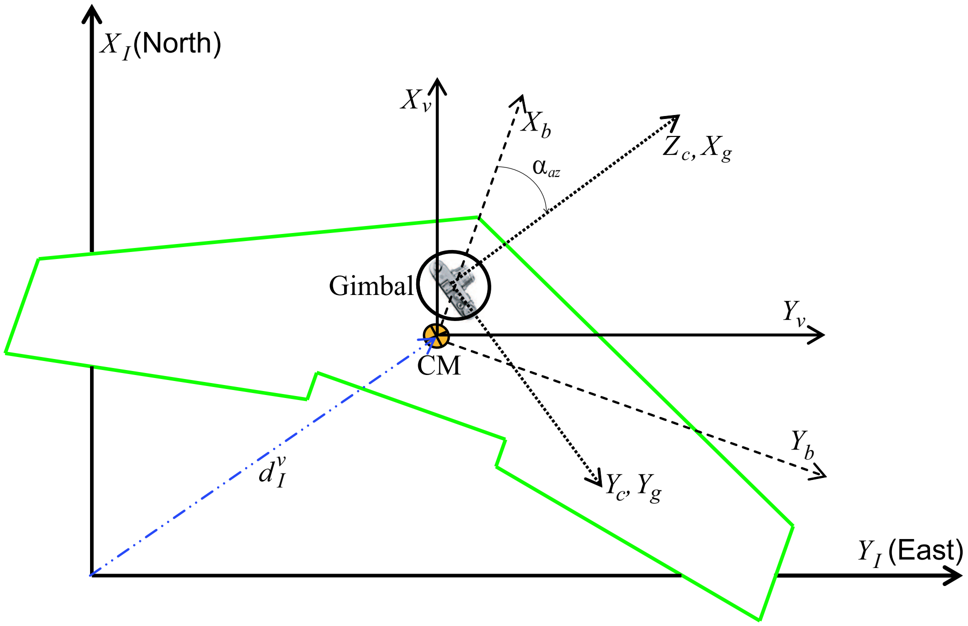


Figure 3.2: Coordinate frames

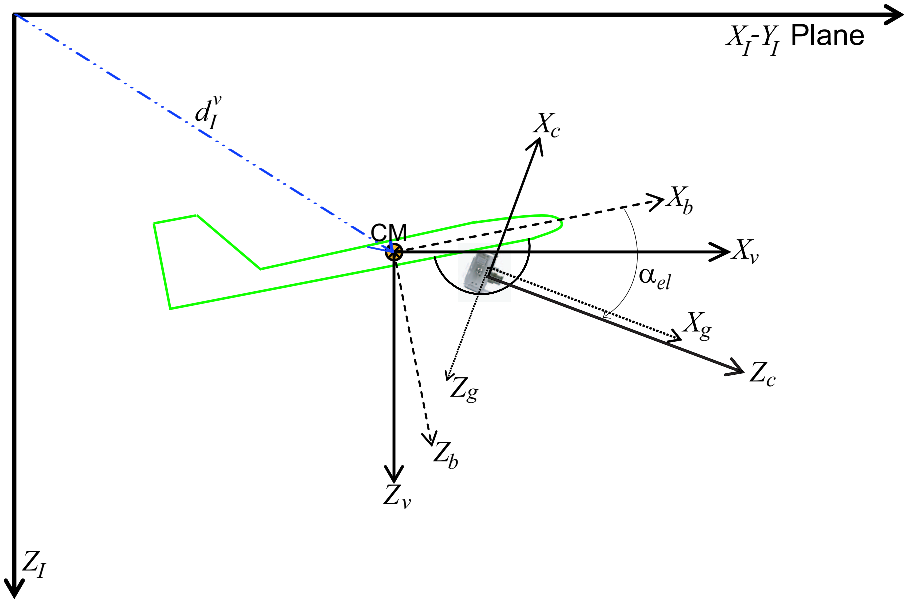


Figure 3.3: Coordinate frames

3.1.2 Problem Outline

The goal of the localization algorithm is to transform an object's pixel location from an image to its inertial location in meters. To give a mathematical overview of this routine, Table 3.2 shows parameters representing the UAV's attitude, inertial location, camera gimbal angles and camera calibration constants.

Table 3.2: UAV and camera parameters

Parameter	Symbol	Units
UAV roll angle	ϕ	rad
UAV pitch angle	θ	rad
UAV yaw angle	ψ	rad
UAV east position	y_{UAV}	m
UAV north position	x_{UAV}	m
UAV altitude	h_{UAV}	m
Gimbal azimuth angle	α_{az}	rad
Gimbal elevation angle	α_{el}	rad
Camera focal length	f	m
Camera X image scaler	S_x	m/pixel
Camera Y image scaler	S_y	m/pixel
Image center X offset	0_x	pixels
Image center Y offset	0_y	pixels

In addition, a (4×4) camera calibration matrix C, and four (4×4) homogeneous transformation matrices (HTMs) are needed to take a target location in pixels to an estimate in the camera frame and then to the inertial frame. A detailed derivation of these HTMs is given in Section 3.2 and C is derived in Section 3.3. Assuming the parameters in Table 3.2 and the HTMs are known or measured for each image captured, the relationship between an object's $[x\ y\ 1\ 1]^T$ pixel location, denoted q, and it's $[x\ y\ z\ 1]^T$ inertial coordinates, denoted p^I_{obj}, can be described by

$$q = CT^c_gT^g_bT^b_vT^v_Ip^I_{\text{obj}}. \tag{3.1}$$

The right hand side of Equation (3.1) (from right to left) consists of the mentioned HTMs, which transform an object's inertial location into an uncalibrated estimate of the object's location in the camera frame. This estimate is then calibrated and scaled using the matrix C, which contains the image's depth, or scaling factor, λ. The problem with Equation (3.1) is that both p^I_{obj} and the λ portion of C are not known. Solving for p^I_{obj} and extracting λ from C, Equation (3.1) becomes

$$p^I_{\text{obj}} = \lambda[CT^c_gT^g_bT^b_vT^v_I]^{-1}q \ , \tag{3.2}$$

which allows for the easy calculation of p^I_{obj}, assuming λ is known. This motivates the need for a method of estimating λ. Before this is accomplished, however, the HTMs and camera calibration matrix should be explained in more detail.

3.2 Transformations

A homogeneous transformation matrix (HTM) combines both rotation and translation between coordinate frames into a single matrix. This section will visit each HTM individually, discussing its elements and purpose. The sub/super-script naming convention for the HTMs in Equations (3.1) and (3.2) is shown by $T^{\text{to}}_{\text{from}}$, and the function of each HTM is described in Table 3.3.

Table 3.3: Homogeneous transformation matrices.

HTM	Description
T^v_I	Transformation from Inertial to UAV Vehicle frame
T^b_v	Transformation from UAV Vehicle to UAV Body frame
T^g_b	Transformation from UAV Body to Gimbal frame
T^c_g	Transformation from Gimbal to Camera frame

The structure of an arbitrary HTM, T^j_i, is shown in Equation (3.3), where R represents a (3×3) rotation matrix and d^j_i represents a (3×1) translation vector. Both rotation and translation occur from the i^{th} to the j^{th} coordinate frame. However, when written in this manner, the rotation occurs first, followed by the translation.

Therefore the translation vector must be resolved in the j^{th} coordinate frame and negated to avoid pre-multiplying it by R_i^j.

$$T_i^j = \begin{bmatrix} & R & & -d_i^j \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.3}$$

3.2.1 Transformation T_I^v

Since the transformation from the inertial to the vehicle frame is actually a single translation, T_I^v will only depend on the UAV's GPS location and barometric altitude measurements shown by

$$\begin{aligned} T_I^v &= \begin{bmatrix} I & -d_I^v \\ 0 & 1 \end{bmatrix} \text{, where} \\ d_I^v &= \begin{bmatrix} x_{\text{UAV}} \\ y_{\text{UAV}} \\ -h_{\text{UAV}} \end{bmatrix} \text{ and} \\ x_{\text{UAV}} &= \text{North location of UAV as measured by GPS} \\ y_{\text{UAV}} &= \text{East location of UAV as measured by GPS} \\ h_{\text{UAV}} &= \text{Altitude of UAV as measured by barometric pressure sensor} \end{aligned}$$

3.2.2 Transformation T_v^b

The transformation from the vehicle frame to the UAV body frame, T_v^b, consists of a single rotation based on measurements of Euler angles, shown by

$$\begin{aligned}
T_v^b &= \begin{bmatrix} R_v^b & 0 \\ 0 & 1 \end{bmatrix}\text{, where} \\
R_v^b &= \text{Rotation from vehicle to body frame} \\
&= \begin{bmatrix} c_\theta c_\psi & c_\theta s_\psi & -s_\theta \\ s_\phi s_\theta c_\psi - c_\phi s_\psi & s_\phi s_\theta s_\psi + c_\phi c_\psi & s_\phi c_\theta \\ c_\phi s_\theta c_\psi + s_\phi s_\psi & c_\phi s_\theta s_\psi - s_\phi c_\psi & c_\phi c_\theta \end{bmatrix} \text{and} \\
\phi &= \text{UAV roll angle} \\
\theta &= \text{UAV pitch angle} \\
\psi &= \text{UAV heading angle} \\
c_* &= \cos(*) \\
s_* &= \sin(*)
\end{aligned}$$

3.2.3 Transformation T_b^g

The transformation from the UAV body to the gimbal frame, T_b^g, will depend on the location of the UAV's center of mass with respect to the gimbal's rotation center. This vector, denoted by d_b^g, is resolved in the gimbal frame. T_b^g will also depend on the rotation that aligns the gimbal's coordinate frame with the UAV's body frame. This rotation is denoted R_b^g and requires measurements of the camera's azimuth and elevation angles α_{az} and α_{el} respectively, both of which are known. This transformation is shown by

$$\begin{aligned}
T_b^g &= \begin{bmatrix} R_b^g & -d_b^g \\ 0 & 1 \end{bmatrix} \text{, where} \\
R_b^g &= R_{y,\alpha_{el}} R_{z,\alpha_{az}} \\
&= \begin{bmatrix} c_{el} & 0 & s_{el} \\ 0 & 1 & 0 \\ -s_{el} & 0 & c_{el} \end{bmatrix} \begin{bmatrix} c_{az} & s_{az} & 0 \\ -s_{az} & c_{az} & 0 \\ 0 & 0 & 1 \end{bmatrix} \\
&= \begin{bmatrix} c_{el}c_{az} & c_{el}s_{az} & s_{el} \\ -s_{el} & c_{az} & 0 \\ -s_{el}c_{az} & -s_{el}s_{az} & c_{el} \end{bmatrix} \text{and} \\
d_b^g &= \text{Vector from gimbal center to cm of UAV },
\end{aligned}$$

where

$$\begin{aligned}
c_* &= \cos(*) \\
s_* &= \sin(*) \\
\alpha_{az} &= \text{Azimuth angle of rotation about } Z_g \\
\alpha_{el} &= \text{Elevation angle of rotation about } Y_g\text{, after } \alpha_{az}.
\end{aligned}$$

3.2.4 Transformation T_g^c

T_g^c is the transformation from gimbal to camera reference frames. It will depend on the vector d_g^c, which describes the location of the gimbal's rotation center relative to the camera center and is resolved in the camera's coordinate frame. T_g^c also depends on a simple rotation R_g^c, which aligns the camera's coordinate frame with that of the gimbal. It is shown by

$$
\begin{aligned}
T_g^c &= \begin{bmatrix} R_g^c & -d_g^c \\ 0 & 1 \end{bmatrix}, \text{ where} \\
R_g^c &= \text{3 dimensional rotation from the gimbal to the camera frame} \\
&= \begin{bmatrix} 0 & 0 & -1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \text{ since we chose } X_c = -Z_g \text{ and } Z_c = X_g \\
d_g^c &= \text{Vector from camera center to gimbal center, resolved in camera frame}
\end{aligned}
$$

We now have four HTMs that are based on a priori calibrations and measurements from on-board sensors. Following Equation (3.2), the next step before estimating a target location is to develop the camera calibration matrix.

3.3 The Calibration Matrix

The image captured from a video feed, as with any image, is a two-dimensional representation of three-dimensional information. The purpose of the calibration matrix is to bring the target's location in pixels into the three-dimensional camera frame in meters. The camera's calibration matrix defines a line from the camera center through the target's location in the image frame, (x_{im}, y_{im}). In inertial coordinates, the target lies some distance λ along this line and can be pin-pointed when λ is known. Simply put, the calibration matrix allows the image to be scaled so that meters can be derived from pixels. Figure 3.1 shows the image frame in pixels (ip) and meters (im). The change of axes between meter and pixel frames is meant to allow easier comparison between image (meter) and camera frames. Trucco, et al [17] show that the change from pixels to meters in the image frame is

$$
\begin{aligned}
x_{im} &= (-y_{ip} + 0_y)S_y \\
y_{im} &= (x_{ip} - 0_x)S_x.
\end{aligned}
\tag{3.4}
$$

Equation (3.4) yields meters in the image frame. Using a perspective camera model, Equation (3.5) extends this to the camera frame and is simply a matrix representation of the law of similar triangles, see Figure 3.1.

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix}_c = \lambda \begin{bmatrix} 1/f & 0 & 0 \\ 0 & 1/f & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}_{im} , \tag{3.5}$$

Combining Equations (3.4) and (3.5), we achieve a relationship between pixels in the image frame and meters in the camera frame. This is shown by

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix}_{ip} = \frac{1}{\lambda} \begin{bmatrix} 0 & \frac{f}{S_x} & 0_x \\ -\frac{f}{S_y} & 0 & 0_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix}_c . \tag{3.6}$$

A closer look at Figure 3.1 reveals that x_{ip} relates to y_c and y_{ip} to x_c in Equation (3.6) since there was a change of axes between the *ip* and *im* coordinate frames. By moving λ to the other side and letting $\frac{f}{S_x} = f_x$ and $\frac{f}{S_y} = f_y$, we achieve

$$\lambda \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}_{ip} = \underbrace{\begin{bmatrix} 0 & f_x & 0_x \\ -f_y & 0 & 0_y \\ 0 & 0 & 1 \end{bmatrix}}_{C} \begin{bmatrix} x \\ y \\ z \end{bmatrix}_c , \tag{3.7}$$

where C represents the camera calibration matrix. This representation shows that λ can be extracted from C as another parameter. More complex calibration matrices exist that account for image skewing and radial and tangential distortions [18], however, with a square CCD array and a narrow angle lens, these can be justifiably ignored without serious consequence.

We now know, or have measurements for, all parameters in Equation (3.2) with the exception of target location, p^I_{obj}, and image depth, λ. Therefore, the next section presents a method of estimating λ.

3.4 Image Depth and Target Location

This section presents a method of generating an estimate for λ using measurements of UAV altitude. It is beneficial to first understand what λ represents and discuss its place in camera model equations before discussing the method further. In computer vision, the term “image depth” refers to the distance along the camera’s optical axis, Z_c, to the object of interest in the image (see Figure 3.1). Its value is often unknown, and is therefore replaced with an arbitrary scalar λ [19]. Solving Equation (3.5) for λ yields

$$\begin{aligned}\lambda &= f\frac{x_c}{x_{im}} \\ &= f\frac{y_c}{y_{im}} \ ,\end{aligned} \tag{3.8}$$

where λ, x_c and y_c are unknown. This equation effectively represents a ray of length λ, whose origin is the camera center and whose direction is defined by the target’s calibrated pixel location (x_{im}, y_{im}). Since Equation (3.8) contains two equations and three unknowns, there are an infinite number of solutions, which motivates the need to estimate λ. However, before λ can be estimated, the center of the camera must be represented relative to world coordinates. To accomplish this, the (3×1) vector needed to describe the camera center in world coordinates is appended with a 1 to create a (4×1) vector, p^I_{cc}, for multiplication with the (4×4) transformation matrices. p^I_{cc} is defined as

$$p^I_{\text{cc}} = \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}^I_{\text{cc}} = [T^c_g T^g_b T^b_v T^v_I]^{-1} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}^c_{\text{cc}} \ , \tag{3.9}$$

where the vector $[x\ y\ z\ 1]^{c\ T}_{\text{cc}}$ is equal to $[0\ 0\ 0\ 1]^T$, since it describes the location of the camera center in camera coordinates. Note the absence of the calibration matrix, C, in Equation (3.9). This is due to the fact that the camera is fixed to the airframe, whose position is known in world coordinates, and its center can be calculated without knowledge of λ.

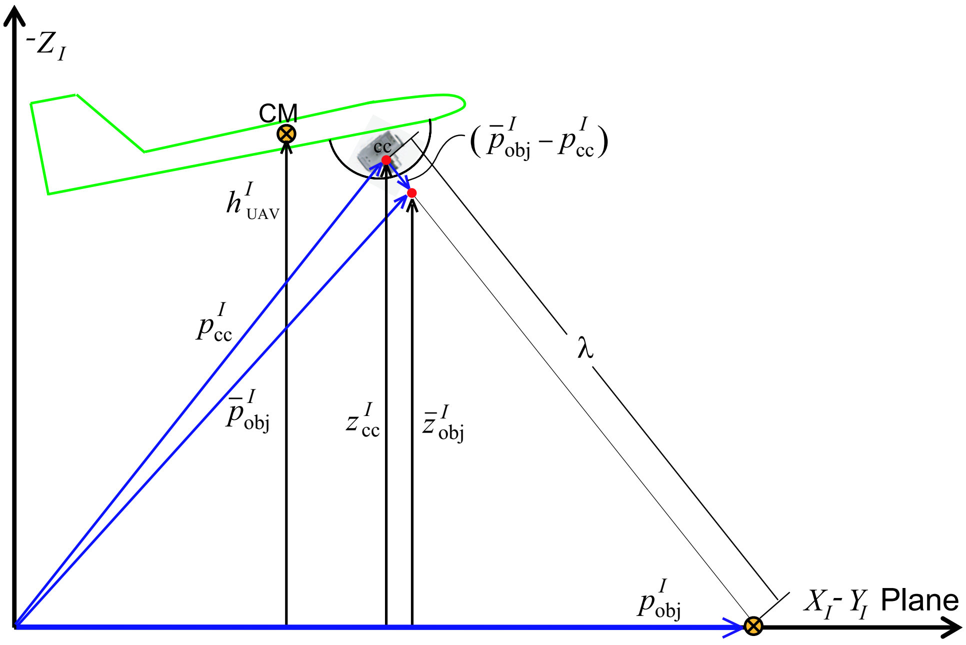


Figure 3.4: Localization vectors

Figure 3.4 shows the camera center, cc, and the vectors describing the target location with respect to both camera and inertial coordinate frames. The vector p^I_{obj} describes the true location of the target in the inertial frame, while $\bar{p}^I_{\text{obj}}$ represents the unscaled target location in the inertial frame. In addition to p^I_{cc}, given in Equation (3.9), $\bar{p}^I_{\text{obj}}$ is also needed to estimate λ, and ultimately p^I_{obj}. Calculating $\bar{p}^I_{\text{obj}}$ requires the target's pixel location, q, and the camera calibration matrix, C, as shown by

$$\bar{p}^I_{\text{obj}} = \begin{bmatrix} \bar{x} \\ \bar{y} \\ \bar{z} \\ 1 \end{bmatrix}^I_{\text{obj}} = [C T^c_g T^g_b T^b_v T^v_I]^{-1} q \;, \tag{3.10}$$

where q is defined as

$$q = [x_{ip} \; y_{ip} \; 1 \; 1]^T \; . \tag{3.11}$$

Referring again to Figure 3.4, the inertial location of the target can be described using the location of the camera center in inertial coordinates, p^I_{cc}, the image depth, λ, and the unscaled location of the target in inertial coordinates, $\bar{p}^I_{\text{obj}}$. This relationship is shown by

$$p^I_{\text{obj}} = \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}^I_{\text{obj}} = p^I_{\text{cc}} + \lambda \left(\bar{p}^I_{\text{obj}} - p^I_{\text{cc}} \right) \ , \tag{3.12}$$

which remains dependent on the image depth, λ. However, recalling the assumption that the target lies on the plane of zero altitude, as calibrated by the UAV, the z components of Equation (3.12) form the relationship

$$0 = z^I_{\text{cc}} + \lambda \left(\bar{z}^I_{\text{obj}} - z^I_{\text{cc}} \right) \ . \tag{3.13}$$

From Equations (3.9) and (3.10), both z^I_{cc} and $\bar{z}^I_{\text{obj}}$ are known, allowing Equation (3.13) to be solved for λ, as shown by

$$\lambda = \frac{-z^I_{\text{cc}}}{\left(\bar{z}^I_{\text{obj}} - z^I_{\text{cc}} \right)} \ . \tag{3.14}$$

Since the inertial Z-axis, Z_I, is defined positive toward the center of the earth, z^I_{cc} will be negative for flight altitudes greater than what was calibrated as zero. Thus, Equation (3.14) yields a positive value for λ, as expected. Now that λ is known, it can be used to approximate the location of the target according to Equation (3.12).

3.5 Filtering

Each image of the target, combined with a measurement of UAV altitude is capable of target localization. However, bandwidth constraints restrict the number of usable images to about five per second. Fortunately, five estimates of target location per second is frequent enough to allow for effective filtering. For this research, several filtering methods were tested, including true average, moving average, recursive least squares and Kalman filters. Each of these filters will be detailed in this section, and the results of hardware tests are presented and discussed in Chapter 5.

3.5.1 True Average

An average, $\bar{x}$ or arithmetic mean, is simply the sum of all the observations, x_i, divided by the number of observations, N, as shown by

$$\bar{x} = \frac{1}{N}\sum_{i=1}^{N} x_i \ , \tag{3.15}$$

where N represents the total number of observations made. The average of a set of normally distributed samples represents the value of maximum likelihood.

3.5.2 Moving Average

A moving average filter is a special case of the average. Instead of using all observations to calculate a mean, only the n most recent are used. The moving average, $\bar{x}_n$ is calculated by

$$\bar{x}_n = \frac{1}{n}\sum_{i=1}^{n} x_i \ , \tag{3.16}$$

where n represents the number of samples used to calculate this average. In effect, data older than n time steps is no longer counted toward the average, hence it "moves" to keep up with the most recent samples. This method needs far less storage space than a true average, but has the tendency to be more affected by outlying data points and is therefore less stable.

3.5.3 Recursive Least Squares

Recursive Least Squares (RLS) is a simple method of recursively fitting a set of points to some function of choice by minimizing the sum of the squares of the offsets of the points. Typically, an RLS algorithm is used to fit a set of points to a characteristic line or quadratic, however, it can be also be used to find a characteristic point as shown in the following table. In this case, the result of the RLS algorithm is identical to the result of the true average, only obtained through a more efficient process.

RLS filter summary

Inputs:

1. Camera center location, $p^I_{\text{cc}} = [x^I_{\text{cc}},\ y^I_{\text{cc}},\ z^I_{\text{cc}},\ 1]^T$
2. Unscaled target location, $\bar{p}^I_{\text{obj}} = [\bar{x}^I_{\text{obj}},\ \bar{y}^I_{\text{obj}},\ \bar{z}^I_{\text{obj}},\ 1]^T$
3. Image depth estimate, λ

Matlab Pseudo-Code for X:

```
persistent $P_N$, $A_N$, $b_N$;
        $a_{N_1} = I_{1\times 1}$    (3.17)
        $b_{N_1} = \left[\ x^I_{\text{cc}} + \lambda(\bar{x}^I_{\text{obj}} - x^I_{\text{cc}})\ \right]$    (3.18)

if( isempty($A_N$) )
        $A_N = [a_{N_1}]$    (3.19)
        $b_N = [b_{N_1}]$    (3.20)
        $P_N = ({A_N}^T A_N)^{-1}$    (3.21)
        $X_{N_1} = P_N {A_N}^T b_N$    (3.22)

else
        $P_{N_1} = P_N - \frac{P_N {a_{N_1}}^T a_{N_1} P_N}{1 + a_{N_1} P_N {a_{N_1}}^T}$    (3.23)
        $A_{N_1} = [A_N\ a_{N_1}]^T$    (3.24)
        $b_{N_1} = [b_N\ b_{N_1}]^T$    (3.25)
        $X_{N_1} = P_{N_1} {A_{N_1}}^T b_{N_1}$    (3.26)

end
        $P_N = P_{N_1}$    (3.27)
        $A_N = A_{N_1}$    (3.28)
        $b_N = b_{N_1}$    (3.29)

return $X_{N_1}$
```

* $I_{1\times 1}$ refers to the (1×1) identity matrix

** The same equations apply for Y, only $b_{N_1} = [y^I_{\text{cc}} + \lambda(\bar{y}^I_{\text{obj}} - y^I_{\text{cc}})]$

3.5.4 Kalman Filter

A Kalman filter recursively calculates an optimal estimate of the states of a dynamic system. This estimate is optimal in the sense that it will have a lower error covariance than any other unbiased, linear estimate. The Kalman filter will fuse the state estimates calculated from the system model with those calculated from system measurements according to the matrices Q and R, respectively. Q represents the expected variance in the system model and R represents the expected variance in the system measurements. In other words, the values in Q and R indicate how much the filter will trust the system model and sensor measurements respectively when calculating individual state estimates. The benefit of this structure is the versatility it affords since both Q and R can be time-varying. In this case, where the target is assumed stationary, the system model has zero dynamics and is considered very accurate. This creates a Q matrix with very small numbers along the diagonal, which represent small variances in the model of the target.

If a moving target is to be localized in future work, the non-zero dynamics of the target model would then be used in place of the previous model and the Kalman filter would adjust its state estimates accordingly. Note that changing the system model also changes the confidence in model predictions, which is represented by Q. Also, in contrast to the averaging routine outlined previously, no record of earlier samples is required to yield a solution. This presents a distinct advantage for implementation on a micro-controller where memory space is limited. The pseudo-code outlining this filter is shown in the table below.

Kalman filter summary

Inputs:

1. Camera center location, $p^I_{\text{cc}} = [x^I_{\text{cc}},\ y^I_{\text{cc}},\ z^I_{\text{cc}},\ 1]^T$
2. Unscaled target location, $\bar{p}^I_{\text{obj}} = [\bar{x}^I_{\text{obj}},\ \bar{y}^I_{\text{obj}},\ \bar{z}^I_{\text{obj}},\ 1]^T$
3. Image depth estimate, λ

Matlab Pseudo-Code:

```
    persistent $x_{old}$, $P_{old}$, $Q$, $R$;
    if( isempty($x_{old}$) )
        $x_{old} = [0,\ 0]^T;$    (3.30)
        $P_{old} = (1200)I_{2\times 2};$    (3.31)
        $Q = (.01)I_{2\times 2};$    (3.32)
        $R = (100)I_{2\times 2};$    (3.33)
    else
    % Q & R could change dynamically here
    end
        $Y = \begin{bmatrix} x^I_{\text{cc}} + \lambda(\bar{x}^I_{\text{obj}} - x^I_{\text{cc}}) \\ y^I_{\text{cc}} + \lambda(\bar{y}^I_{\text{obj}} - y^I_{\text{cc}}) \end{bmatrix}$    (3.34)
    % update $P$, find $K$, update $x$ & $P$, propagate $x$ & $P$
        $P = P_{old} + Q;$    (3.35)
        $K = P(P + R)^{-1};$    (3.36)
        $x = x_{old} + K(Y - x_{old});$    (3.37)
        $P = (I_{2\times 2} - K)P;$    (3.38)
        $x_{old} = x;$    (3.39)
        $P_{old} = P;$    (3.40)

    return $x$
```

*where $I_{2\times 2}$ refers to the (2×2) identity matrix

3.6 Summary

The goal of this research was to determine the inertial location of a stationary, ground-based target from a fixed-wing UAV using computer vision. The mathematics required to accomplish this were presented and thoroughly discussed. The first piece of information needed for localization is the target's pixel location in an image taken by the overhead UAV. This location, combined with knowledge of the UAV's position, attitude and camera pose angles at the time the image was taken, as well as the camera's intrinsic properties, provide an unscaled estimate for the inertial location of the target. When the scaling parameter, or image depth, λ, is known, this location can be scaled into a meaningful estimate of actual target position in world coordinates. Since a pixel location and image depth exist for each image taken of the target, several filtering methods are used to combine each new estimate into a single location.

Chapter 4

Error Analysis

Current research in localization uncertainty and error propagation is typically restricted to ground-based mobile robots [20, 21, 22, 23]. This chapter will present a study of uncertainties in the parameters of UAV flight and how these propagate into errors in target localization. Like all aircraft, UAVs are susceptible to outside influences. These include wind gusts, jet streams, variations in atmospheric pressures, air densities and temperatures. These phenomena, among others, add unwanted noise to aircraft sensors. This noise, combined with sensor errors and inaccuracies, contaminates each measurement of position, altitude, airspeed and heading as well as roll, pith and yaw rates. The purpose of this chapter is to explore the main error sources in UAV and gimbal control and to study how each one individually affects the target localization result. First, the main error sources are discussed in Section 4.1. Section 4.2 then studies how sensitive the estimated target location is to each source and determines which are most significant. Finally, Section 4.3 discusses methods of minimizing these major errors and ways in which to handle noise in general.

4.1 Error Sources

In the equation

$$p^I_{\text{obj}} = \lambda[CT^c_gT^g_bT^b_vT^v_I]^{-1}q, \tag{4.1}$$

each term introduces inaccuracies to the end result. Since λ is calculated from measurements of UAV altitude, its associated errors will be accounted for through altitude uncertainty. Errors in the camera calibration matrix, C, are comparatively small since they originate in the calibration routine itself and are therefore neglected. T_g^c depends on the location of the UAV center of mass with respect to the camera center. The errors associated with this measurement are on the order of millimeters and can also be ignored. T_b^g on the other hand, will depend on camera gimbal angles, which are not known. The camera gimbal is controlled via commercially available hobby servos, for which no feedback exists. Thus, the commanded camera angles are known, while actual angles are not. Despite this, laboratory tests have shown servo errors to be accurate to less than half a degree and precise to less than one fifth of a degree. Though impressive, these numbers are only valid when the servos are given sufficient time to reach their desired angles. The time required to do so is on the order of 5 ms/deg, which should be sufficiently fast for typical changes in desired angles.

T_v^b introduces further inaccuracies through errors in UAV attitude estimation. Euler angles ϕ, θ, and ψ are calculated from gyro measurements of roll, pitch and yaw rates as well as accelerometer readings with reference to the gravity vector [24, 25]. Unfortunately, gyros tend to drift, causing accumulating errors in ψ. Estimates of ϕ and θ are generated by subtracting the gravity vector from accelerometer measurements, a technique which works well under static conditions. However, this subtraction yields corrupted results when the UAV is experiencing accelerations, a dynamic condition quite common during flight. Due to these limitations, Euler angle estimates are considered decent at best. Through laboratory tests, ϕ, θ and ψ have been shown to be statically accurate to within five degrees, and dynamically accurate to within ten degrees. When the UAV is in a large orbit, static conditions can be assumed.

The translation from inertial to vehicle frames, accomplished by T_I^v, adds inaccuracies that stem from both GPS measurements and barometric altitude readings. The major GPS inaccuracies are attributed to a variety of sources that combine to achieve an accuracy of roughly 10 m in the horizontal plane, and 25 m in the vertical plane [26]. However, since the bias portion of this error equally effects both UAV and target, it can be neglected leaving only random errors, which can measure up to roughly 5 m in the horizontal plane. With the addition of an absolute pressure sensor, altitude inaccuracies are reduced to roughly 8 m [16]. Although altitude errors have a direct influence on the estimation of image depth, as discussed in Section 3.4, it is intuitively expected that target localization will be most sensitive to inaccuracies in camera and UAV attitude angles.

Table 4.1: Uncertainties, U_*

Source	± Value
α_{az}	.5 deg
α_{el}	.5 deg
ϕ	5 deg
θ	5 deg
ψ	5 deg
x_{UAV}	5 m
y_{UAV}	5 m
h_{UAV}	8 m
x_{ip}	5 pixels
y_{ip}	5 pixels

The target pixel location, q, is determined by IDC and is subject to the adverse effects of visual occlusions and lighting changes. Accounting for these error sources, it is believed that q accurately represents the target location in the image within 5 pixels in both x_{ip} and y_{ip}. Although actual uncertainties are not known, the values shown in Table 4.1 are the results of laboratory tests and it is assumed that they represent a 95% probability.

4.2 Sensitivities

For this research, the process and result of estimating a target's inertial location is referred to as target localization. In this section, a study of localization sensitivity to uncertainties in measurements of UAV location and attitude as well as camera gimbal angles is presented. For the purpose of experiment, a simulation was created in which all UAV sensors yielded perfect signals with zero error. These perfect measurements were taken from nominal flight conditions listed in Table 4.2 and were used to form the HTMs of Equation (4.1). The camera was placed at

Table 4.2: Nominal UAV conditions for localization,

Source	± Value
α_{az}	90 deg
α_{el}	30 deg
ϕ	20 deg
θ	5 deg
ψ	0 deg
x_{UAV}	0 m
y_{UAV}	0 m
h_{UAV}	60 m
x_{ip}	320 pixels
y_{ip}	240 pixels

90 deg azimuth and 30 deg elevation angles and the UAV was placed 60 m over the home position flying a 0 deg heading with 20 deg roll and 5 deg pitch angles. The target was assumed to be located at the image center, and with all other variables of Equation (4.1) known, finding the target location, p^I_{obj}, became a simple calculation. This location was considered truth and was then compared to the target locations calculated as each of the parameters in Table 4.2 was incrementally perturbed by plus and minus its uncertainty, while all others were held constant. Figures 4.1 through 4.3 show the error from truth as a function of uncertainty in each of these parameters. As expected, increasing uncertainty results in increasing localization error. Figure 4.1 shows the sensitivity of target localization to uncertainties in UAV attitude and camera gimbal angles.

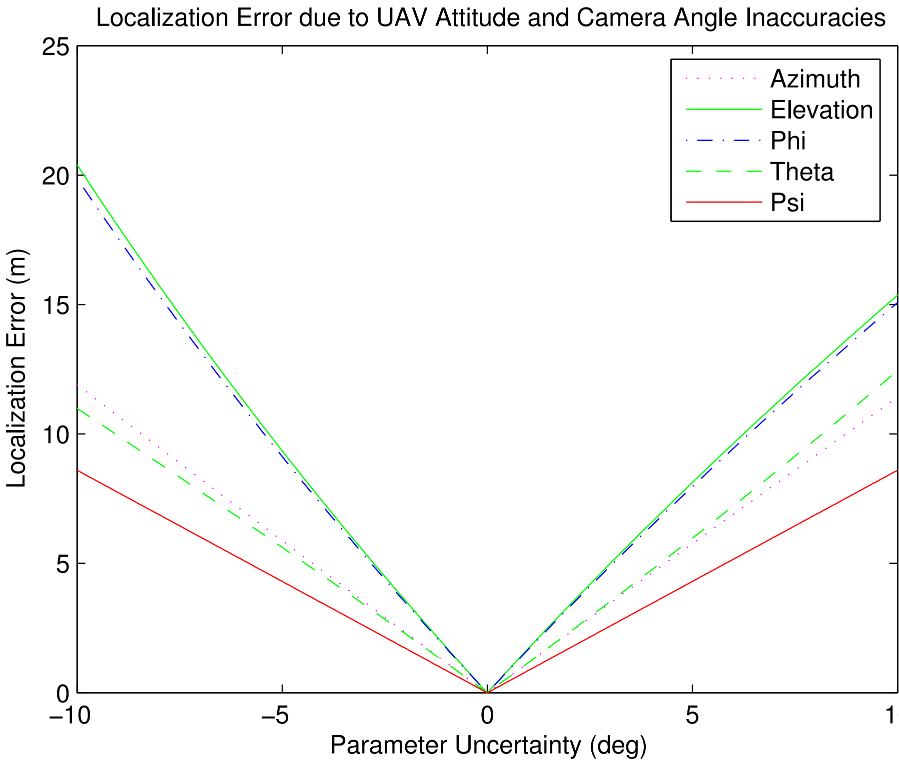


Figure 4.1: Sensitivity of localization to uncertainties in camera and UAV attitude angles

As shown in Figure 4.1, good estimates of UAV roll angle and camera elevation angle are important since the estimated target location is most sensitive to deviations in these parameters. The fact that localization is equally sensitive to both UAV roll and camera elevation angles is expected since during a localization flight the camera is panned to roughly ninety degrees, which aligns Y_g, the axis about which elevation occurs, nearly parallel to X_b, the axis about which aircraft roll occurs. This alignment means the localization algorithm cannot differentiate between changes in elevation angle and changes in UAV roll angle.

$$p_{\text{obj}}^{I} = \lambda[CT_g^c T_b^g T_v^b T_I^v]^{-1} q = F(\alpha_{az}, \alpha_{el}, \phi, \theta, \psi, (x, y, h)_{\text{UAV}}) \tag{4.2}$$

Since the sensitivity plots from Figures 4.1, 4.2 and 4.3 are linear over the expected error range, the partial derivative of target location F, Equation (4.2), with

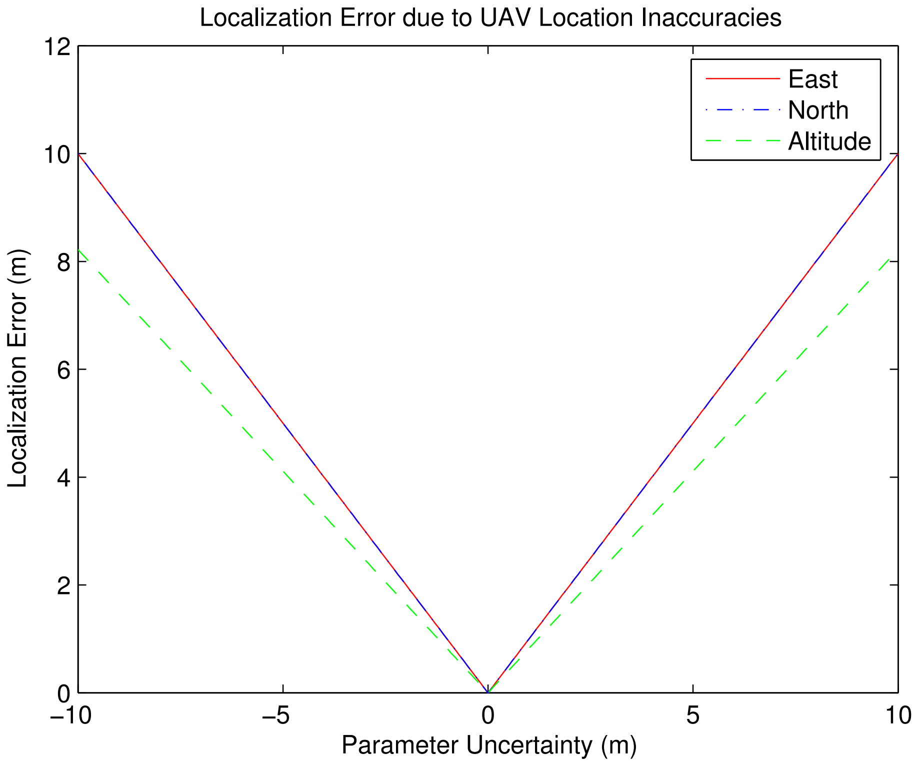


Figure 4.2: Sensitivity of localization to uncertainties in UAV location

respect to each error source can be estimated as simply the slope of each curve. For increased confidence in these partials, the method of Sequential Perturbation (SP) discussed in [27] was also used and the results of both methods are tabulated in Table 4.3.

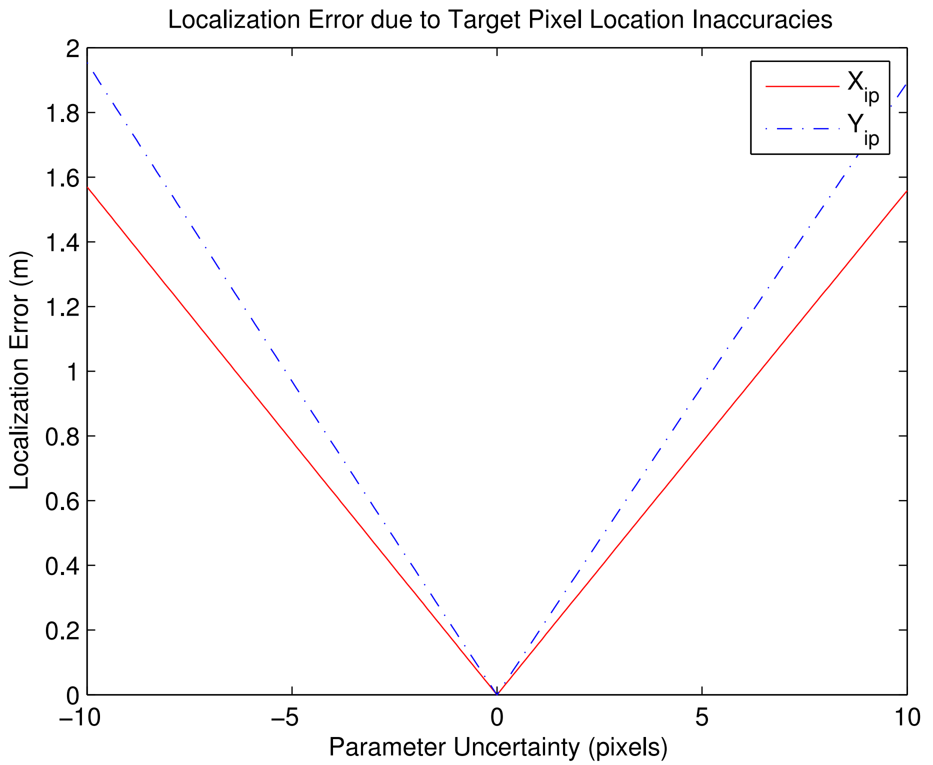


Figure 4.3: Sensitivity of localization to uncertainties in target pixel location

Table 4.3: Estimated partial derivatives, $\frac{\partial F}{\partial *}$

i	Source	Slope	SP
1	α_{az}	1.2 m/deg	1.1 m/deg
2	α_{el}	2.0 m/deg	1.7 m/deg
3	ϕ	2.0 m/deg	1.7 m/deg
4	θ	1.1 m/deg	1.1 m/deg
5	ψ	0.8 m/deg	0.8 m/deg
6	x_{UAV}	1.0 m/m	1.0 m/m
7	y_{UAV}	1.0 m/m	1.0 m/m
8	h_{UAV}	0.8 m/m	0.8 m/m
9	x_{ip}	0.2 m/pixel	0.15 m/pixel
10	y_{ip}	0.16 m/pixel	0.19 m/pixel

Using Table 4.1 in conjunction with the partials from the slope method in

Table 4.3, it is possible to determine the expected localization error, Γ, resulting from the combination of all error sources [27]. For example, Table 4.1 shows that the uncertainty in UAV roll angle, U_ϕ, is 5 deg. From Table 4.3, the partial derivative of target location with respect to UAV roll, $\frac{\partial F}{\partial \phi}$, is approximately 2 m/deg. This means 5 deg in roll uncertainty yields roughly 10 m of error in the estimated target position, which is added into the total localization uncertainty by

$$\begin{aligned}\Gamma &= \sqrt{\sum_{i=1}^{N}\left(\frac{\partial F}{\partial i}U_i\right)^2} \\ &= 15.5 \text{ m Slope} \\ &= 14.6 \text{ m Sequential Perturbation} \ , \end{aligned} \tag{4.3}$$

where i refers to each of the N parameters on which F is dependent, as shown in Equation (4.2) and listed in Table 4.3. By applying the appropriate values found in Tables 4.1 and 4.3, it is theoretically possible to locate a target within 15.5 m using computer vision from a fixed-wing UAV at an altitude of 60 m.

4.3 Error Minimization

Now that a reasonable grasp of expected errors is obtained, the logical question that follows is: What can be done to minimize them? The solution to the problem underlying this question involves acquiring more accurate UAV information, especially for roll angle and altitude. Improvements in these two values would result in a large increase in overall accuracy since target localization is very sensitive to θ and the uncertainty in altitude is high. For example, if UAV roll was known to 3 deg, rather than 5 deg, and altitude was known to within 5 m, rather than 8 m, the localization accuracy would theoretically drop from 15.5 m to 12.2 m. For this research, dealing with the error sources directly means a careful calibration of KAP accelerometers, rate gyros, pressure sensors and camera gimbal angles before each flight. However, the practical answer to the question has more to do with minimizing the effects of these errors rather than minimizing the errors themselves.

To minimize the effects of these uncertainties, localization estimates are generated from multiple vantage points as the UAV orbits over the target. The benefit of multiple estimates is that their variation can be characterized by a normal distribution about an estimated target location [27]. As the number of estimates increases, the uncertainty of this estimated location decreases. As the uncertainty decreases, the estimated location will approach the true target location. In other words, each additional result will tend the mean of all results toward its central value, the true target location. However, this assumes that each estimate is unbiased, and since this is intuitively doubtful, even the filtering methods discussed in section 3.5 will be subject to bias errors.

Chapter 5

Results

The focus of this research is the vision-based localization of a ground-based target when imaged from a fixed-wing UAV. Since an image is two-dimensional in nature, its depth, or λ, must be estimated. Chapter 3 of this thesis presented a method of estimating this dimension, and showed how this estimate leads to a target location in world coordinates. Since a λ, and thereby an estimated target location, was calculated specific to each image of the target captured, Chapter 3 also discussed several methods of combining these locations through filters designed to cancel noise. This chapter will present the results of actual tests run in hardware to localize a target from a small, fixed-wing UAV.

The results of an initial hardware experiment are shown in Figure 5.1. The plot shows actual, estimated and filtered target locations from a UAV flying at 60 m altitude in a 50 m radius circular orbit around an initial guess of the target location.

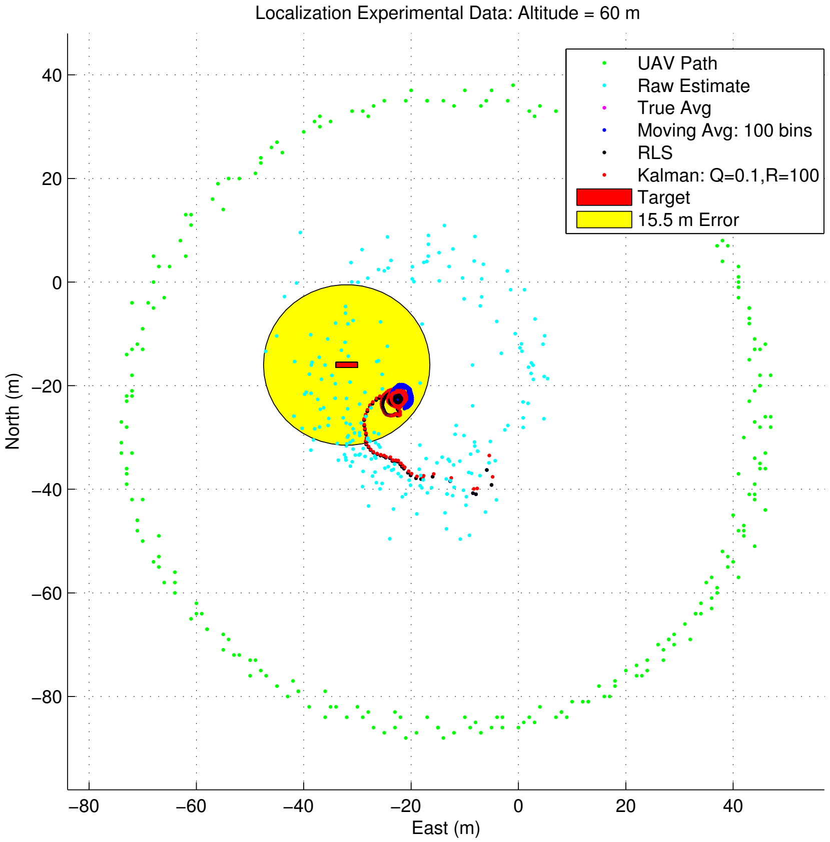


Figure 5.1: Localization results

As can be seen in Figure 5.1, the majority of the individual estimates are not within the expected accuracy range of 15.5 m derived in Chapter 4. To improve these results, the effect of factors such as steady winds should be considered and accounted for. Even a mild wind can introduce significant errors in UAV heading and roll. The term UAV heading is used rather loosely here since there is currently no accurate method of measuring true heading. Instead, a GPS heading is measured, which provides the ground course rather than the true heading of the UAV. The difference between heading and ground course can be significant, especially when winds are perpendicular to the aircraft's direction of travel. An aircraft traveling crosswind must adjust its true heading slightly into the wind, or “crab”, to maintain a desired ground course. This crabbing causes the aircraft's heading and direction of travel to be misaligned, resulting in higher uncertainty in ψ. As discussed in Chapter 4, this uncertainty will propagate into the estimated target location, causing degraded results. To compensate for wind, the data from the same experiment was post-processed, disregarding all target location estimates generated when the UAV was traveling perpendicular to the 5 mph wind. This resulted in the convergence of the estimated target location within 10.9 meters of the actual target location, as seen in Figure 5.2.

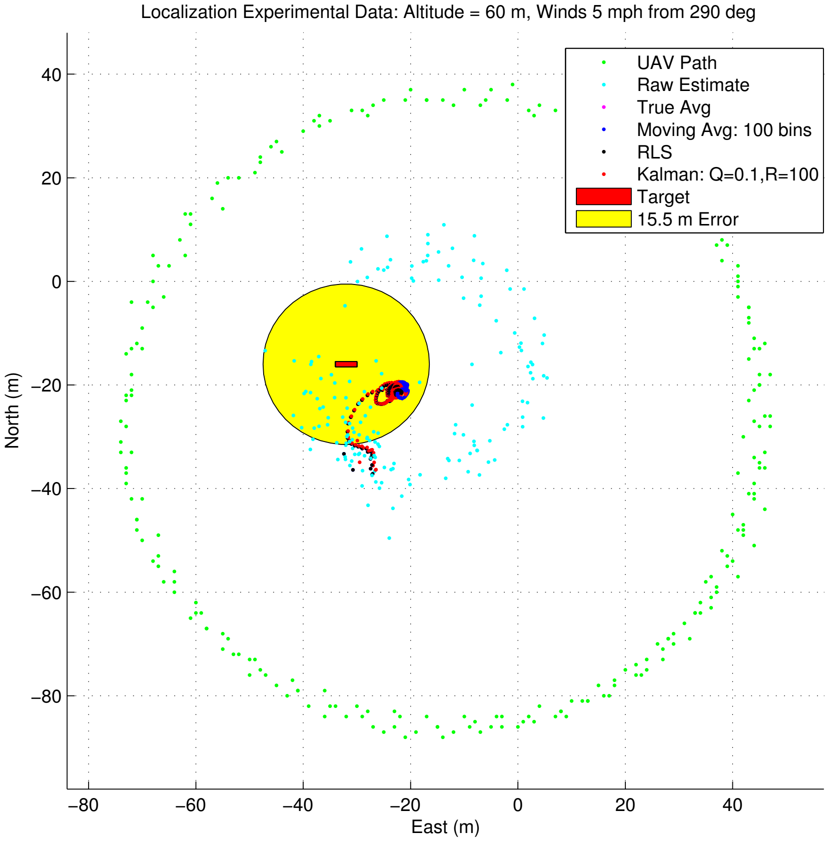


Figure 5.2: Localization results accounting for wind

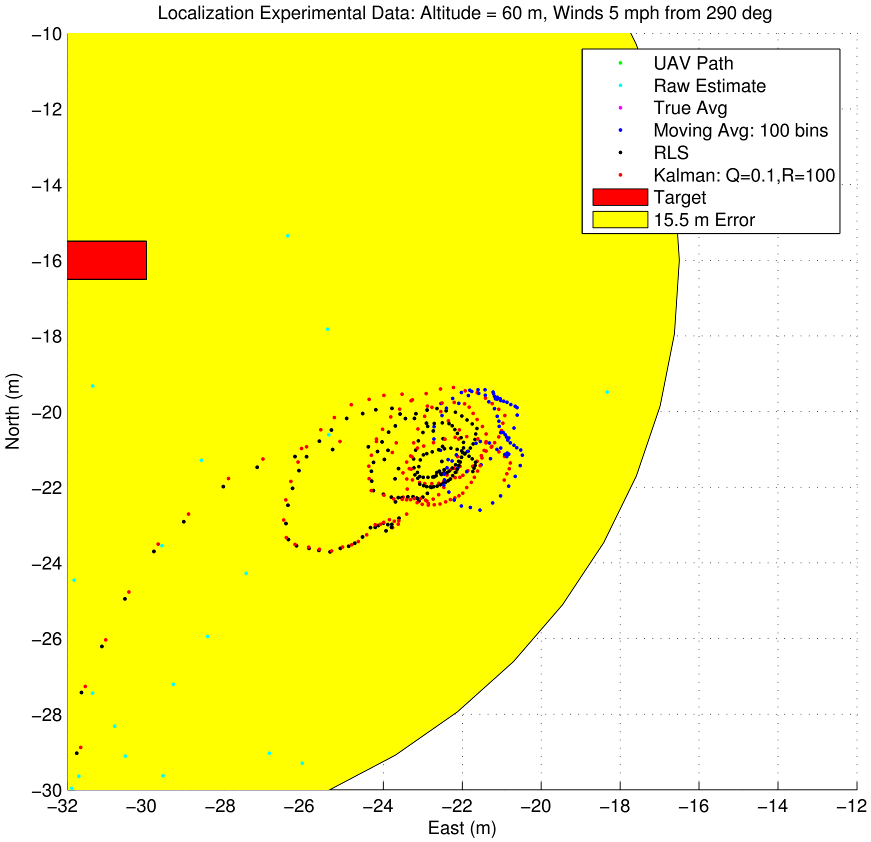


Figure 5.3: Closeup of localization results

Figure 5.3 shows a closer look at the region of convergence and shows the pattern of the filtered estimates. As expected, the true average and RLS filters provide the most stable estimate to target location, however, they also require the most calculations and storage space. A moving average, on the other hand, is more easily affected by a single estimate and therefore never really settles in on a value. With the Kalman filter, accuracy is only slightly sacrificed for a gain in computational storage efficiency. Table 5.1 shows the values on which each filter converged, with and without wind correction.

Table 5.1: Errors in filtering methods

Method	Without Wind Correction	With Wind Correction
True Avg	11.5 m	10.9 m
Moving Avg	12.0 m	11.9 m
RLS	11.5 m	10.9 m
Kalman	12.2 m	11.6 m

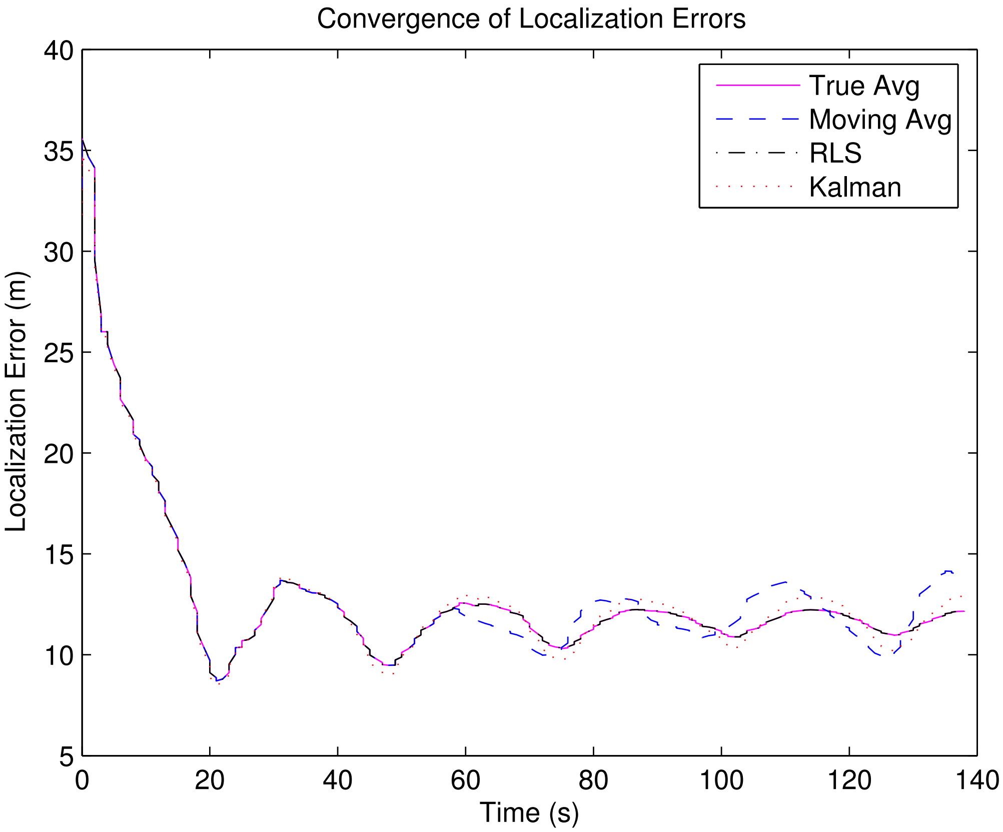


Figure 5.4: Filter error convergence

Figure 5.4 gives an idea of how fast these filters settle on a value for target location. As shown, the true average data is occluded by the RLS data since they are mathematically identical for the estimation of a characteristic point when given a sequence of measured points. A moving average has the benefit of keeping only the data from the most current complete orbit over the target, however, as shown in Figure 5.4 its results do not settle as well over time. Recursive least squares (RLS) finds the point that minimizes the sum of the squares of the distances to each target location estimate and, in this case, is a true average. As shown, the Kalman filter also settles quickly, and although it yields slightly more exaggerated results over the orbit time, it requires minimal storage since only the covariance matrix is propagated, making it a viable choice.

It is interesting to note that none of the filters converge near zero localization error. This indicates that the estimates entering the filters were biased. Although the bias likely consists of multiple, unkown sources, it can be estimated by the error associated with the value at which the filter settles out, which is just over 10 m according to Figure 5.4.

Chapter 6

Conclusions and Future Work

6.1 Conclusions

This thesis shows that a ground-based target can be accurately localized from an overhead UAV using a vision-based approach in real-time. The localization is accomplished from a fixed wing UAV at an altitude of 60 m in orbit of 50 m in radius. The resulting estimate is within 11 m of actual GPS target location after approximately 50 samples. This shows that when UAV altitude is known with reasonable accuracy, the image depth, λ can be reliably estimated. Overall, the hardware results demonstrate the successful implementation of a very practical technology capable of increasing the efficiency of tasks such as reconnaissance, search and rescue and many others.

6.2 Future Work

Although the initial results of hardware testing are very promising, there is still room for improvement. Suggestions for accomplishing this improvement include such ideas as using stereo vision to create an estimate for image depth and attempting to decrease the latency in the sensor measurements used for localization. Implementing these items would provide useful comparison and could serve to increase the accuracy and reliability of vision-based target localization from a fixed wing UAV.

Other methods of estimating image depth exist, and a more thorough study would provide a useful and interesting comparison on localization accuracy. For example, the use of stereo vision to estimate λ would eliminate the need to assume that the target lies on the plane of zero-altitude, effecting localization results.

The problem of sensor latency stems from the fact that sensors are sampled at various times, placing these sampled values into variables that are then used simultaneously in the calculation of target location. Particularly, the image data supplied from IDC significantly lags the data sampled from on-board sensors. Reconciling this disparity could yield a more accurate convergence of the localization algorithm, motivating further study of the issue.

Bibliography

[1] AirForce-Technology, "Projects," 2005, http://www.airforce-technology.com/projects/.

[2] Office of the Secretary of Defense, Ed., *Unmanned Aerial Vehicles Roadmap 2002-2027*, United States Government, Washington DC, USA, 2002.

[3] Luiz Chaimowicz, Ben Grocholsky, James F. Keller, Vijay Kumar, and Camillo J. Taylor, "Experiments in Multirobot Air-Ground Coordination," in *Proceedings of the 2004 International Conference on Robotics and Automation*, New Orleans, LA, April 2004, pp. 4053–4058.

[4] Rene Vidal and Shankar Sastry, "Vision-Based Detection of Autonomous Vehicles for Pursuit-Evasion Games," in *IFAC World Congress on Automatic Control*, Barcelona, Spain, July 2002.

[5] Rene Vidal, Shahid Rashid, Cory Sharp, Omid Shakernia, Jin Kim, and Shankar Sastry, "Pursuit-Evasion Games with Unmanned Ground and Aerial Vehicles," in *Proceedings of the International Conference on Robotics and Automation*, Seoul, Korea, May 2001.

[6] Reed S. Christiansen, "Design of an Autopilot for Small Unmanned Air Vehicles," M.S. thesis, Brigham Young University, Provo, Utah 84602, August 2004.

[7] Anibal Ollero and Luis Merino, "Control and Perception Techniques for Aerial Robotics," in *7th IFAC Symposium on Robot Control (SYROCO 2003)*, Wroclaw, Poland, September 2003.

[8] B. Barshan and H. F. Durrant-Whyte, “Orientation Estimate for Mobile Robots Using Gyroscopic Information,” in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, Munich, Germany, September 1994, pp. 1867–1874.

[9] P. Saeedi, D. G. Lowe, and P. D. Lawrence, “3D Localization and Tracking in Unknown Environments,” in *Proceedings of the IEEE Conference on Robotics and Automation*, Sept 2003, vol. 1, pp. 1297–1303.

[10] S. G. Chroust and M. Vincze, “Fusion of Vision and Inertial Data for Motion and Structure Estimation,” *Journal of Robotic Systems*, vol. 21, pp. 73–83, January 2003.

[11] Horst-Michael Gross, Hans-Joachim Boehme, and Torsten Wilhelm, “A Contribution to Vision-based Localization, Tracking and Navigation Methods for an Interactive Mobile Service-robot,” in *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, October 2001, vol. 2, pp. 672–677.

[12] Thomas Rofer and Matthais Jungel, “Vision-based Fast and Reactive Monte-Carlo Localization,” in *Proceedings of the IEEE International Conference on Robotics and Automation*, September 2003.

[13] Carlos F. Marques and Pedro U. Lima, “Multi-sensor Navigation for Soccer Robots,” in *RoboCup 2001: Robot Soccer World Cup V*, 2001, vol. 2377 of *Lecture Notes in Computer Science*, pp. 144–153.

[14] Rolf Rysdyk, “UAV Path Following for Constant Line-of-sight,” in *2nd AIAA “Unmanned Unlimited” Systems, Technologies and Operations Aerospace, Land and Sea Conference*, September 2003.

[15] Sebastian Stolle and Rolf Rysdyk, “Flight Path Following Guidance for Unmanned Air Vehicles with Pan-Tilt Camera for Target Observation,” in *22nd Digital Avionics Systems Conference*, October 2003.

[16] Procerus Technologies, “Procerus UAV,” 2005, http://www.procerusuav.com.

[17] Emanuele Trucco and Alessandro Verri, *Introductory Techniques for 3-D Computer Vision*, Prentice-Hall, New Jersey, USA, 2002.

[18] Jean-Yves Bouguet, "Camera Calibration Toolbox for Matlab," 2004, http://www.vision.caltech.edu/bouguetj/calib_doc/index.html.

[19] Yi Ma, Stefano Soatto, Jana Kosecka, and S. Shankar Sastry, *An Invitation to 3-D Vision From Images to Geometric Models*, Springer-Verlag, New York, USA, 2003.

[20] Alonzo Kelly, "General Solution for Linearized Error Propagation in Vehicle Odometry," in *Proceedings of the 10th International Symposium of Robotics Research (ISRR 2001)*, November 2001.

[21] Stergios Roumeliotis and Ioannis M. Rekleitis, "Analysis of Multi-robot Localization Uncertainty Propagation," in *Proceedings of the International Conference on Intelligent Robots and Systems (IROS 2003)*, October 2003.

[22] Carlos Santo, Alain Diou, and Yvon Voisin, "Analysis and Performances of an Object Localization and a Dimensional Measurement Method Applied to the Calibration of Cameras," in *Proceedings of the 22nd International Conference on Industrial Electronics, Control and Instrumentation (IECON 1996)*, August 1996, vol. 2, pp. 790–795.

[23] Stephen Se, David Lowe, and Jim Little, "Local and Global Localization for Mobile Robots using Visual Landmarks," in *Proceedings of the Conference on Intelligent Robots and Systems (IROS 2001)*, 2001, vol. 1, pp. 414–420.

[24] Derek B. Kingston, "Implementation Issues of Real-Time Trajectory Generation on Small UAVs," M.S. thesis, Brigham Young University, Provo, Utah 84602, April 2004.

[25] Derek B. Kingston and Randal W. Beard, "Real-Time Attitude and Position Estimation for Small UAVs Using Low-Cost Sensors," in *AIAA 3rd Unmanned Unlimited Systems Conference and Workshop*, Chicago, Il, September 2004.

[26] Montana State University, "GPS Accuracy," 2004, http://www.montana.edu/places/gps/lres357/slides/GPSaccuracy.ppt.

[27] Richard S. Figliola ad Donald E. Beasley, *Theory and Design for Mechanical Measurements*, John Wiley and Sons, Inc., New York, USA, 2000.