IMAGE-BASED VISUAL SERVOING OF A VTOL UAV

by

Jahanzeb Tariq Khan

Under the supervision of Dr. Abdelhamid Tayebi

A Thesis Submitted in Partial Fulfillment
of the Requirements for the Degree of

Master of Science
in Control Engineering

Lakehead University, Thunder Bay, Ontario, Canada

August 2009

ABSTRACT

Unmanned Aerial Vehicles (UAVs) are an area of significant research activity these days because of their use in dull, dangerous, and dirty tasks. Lack of human pilot paves the way for advanced and more sophisticated control algorithms that should just not control each and every aspect of flight dynamics but also accomplish the desired task at the same time. Visual Servoing is one such way of controlling the attitude and position of aircrafts by using camera as a sensor.

Visual servoing is of two types: image-based visual servoing (IBVS) and position-based visual servoing (PBVS). In IBVS approach, only one camera is used to control the motion of a robot while in PBVS approach two cameras are used to guide a robot. Using PBVS approach, it is possible to obtain 3D coordinates of objects in the environment using epipolar geometry but that is not the case with IBVS approach. In IBVS approach, the depth of the object is lost due to the fact that a 3D scene is being projected on a 2D image surface. To overcome this problem, the technique of spherical projection was developed by Hamel and Mahony in 2002. In most applications of IBVS algorithms to date, one common assumption is of target being stationary which however may not be the case in certain practical applications like crime-fighting or cinematography. An attempt is made to tackle this kind of problem and an IBVS controller is designed so that the UAV can chase a moving target.

A computer aided design (CAD) model of a Vertical Take-Off and Landing (VTOL) type UAV was developed for experimental purposes. The dynamical model of the aircraft is manipulated to facilitate the control design. The dynamics of error are modeled from first principles and then two IBVS controllers are proposed. The first IBVS controller design is based on the assumption that the target velocity is known, while the second one deals with unknown target velocity.

The control strategy (in both cases) relies on the design of driving force for the translational dynamics, from which the desired orientation and thrust are extracted. Thereafter, an inner high gain feedback loop is used to generate the control torque guaranteeing the convergence of the actual attitude to the desired one.

Simulation results are provided to illustrate the effectiveness of both controllers.

ACKNOWLEDGEMENT

After a pretty tiresome, stressful, and frustrating first semester of my undergraduate degree, when I was about to quit engineering, Dr. Krishnamorthy Natarajan's motivating remarks brought me back on track and helped me realize my potential. Ever since then he has been a great mentor as well as a sincere advisor to me and has provided me with valuable suggestions that helped me a lot to take critical decisions about my future. I would sincerely like to express my gratitude with the bottom of my heart for his valuable comments, suggestions, and advices, as well his invitation for doing graduate studies.

The other lovely professor for whom I would like to express my gratitude is Dr. Xiaoping Liu, who along with Dr. K. Natarajan persuaded me a lot to pursue graduate studies.

I would also like to thank Andrew Roberts for his tutorials on how to use Autodesk Inventor as well as his help for designing electrical boards for the project.

Lastly, I am also very grateful to my supervisor Dr. Abdelhamid Tayebi for the reason not just that he supervised me but also let me work on the UAV project. Thanks to his reference letter along with the thesis topic, I was able to get admission for my postgraduate studies in world's premier aerospace school, i.e. UTIAS. For an ordinary student like me, getting admission in UTIAS is very exceptional and I have no words to thank him. I will always be indebted for his favors.

Contents

List of Figures

Chapter 1

INTRODUCTION

The notion of aircraft control is not new and the quest to make autonomous aircraft has been an ongoing challenge for scientists and engineers working in aviation industry. Aircraft control using control surfaces is vital to air flight. Legend has it that the primitive attempts to fly failed mainly due to poor control of the control surfaces. Indeed the primary achievement of Wright Brothers' is the development of practical flight control systems which resulted in mankind's dream of flying come true. Today with advances in technology, the control of aircraft is not just limited to controlling control surfaces; it is rather a sophisticated task with different subsystems (such as engine control, propeller control, etc.) controlling different operations of the aircraft.

Aircraft of varying shapes, sizes, and capabilities are used for different purposes and tasks. In some cases, the task that needs to be performed can be dangerous in the sense that it may require close proximity flights (for ex. electrical power line inspection for fault tolerance) or the nature of the task is too delicate (for ex. crime scene observation). In such a scenario, the life of the pilot can be put at risk and the need for unmanned aircraft arise.

Unmanned aircraft or **Unmanned Aerial Vehicles** (UAV) are an area of significant research activity these days. Lack of human pilot paves the way for advanced and more sophisticated control algorithms that should just not control each and every single aspect of flight dynamics but also accomplish the desired task at the same time. Visual Servoing, described below, is one such way of controlling the attitude and position of aircraft by using camera as a sensor.

1.1 VISUAL SERVOING

UAV control algorithms rely on the information gathered through different sensors such as Inertial Measurement Unit (IMU), Global Positioning System

(**GPS**), etc. While the information gathered through these sensors would be appropriate to make a UAV autonomous if the environment is known a priori and/or is inert, it would pose limitations in case if it is not. Occasionally, for certain tasks the desired target position may be defined relative to particular object or landmark in the environment, and in such cases the traditional positioning equipment, such as a GPS, is prone to failures or malfunction. One cost effective solution to problems of this nature is the use of camera as a sensor to extract the information about the environment or the target position. Such a technique in which computer vision is used to control the motion of a robot is called **visual servo control** or simply just **visual servoing**. This area of robotics which emerged in early 1970s relies on techniques from three areas, namely computer vision, image processing, and control theory.

Two types of visual servo controllers can be designed, either Image-Based or Position-Based. In position-based control, the extracted features are used along with the geometric model of the target to determine the pose of the target with respect to the camera. Image-based approach uses the same procedure but instead of reconstructing the target position, the servoing is done directly on image features. These visual servo schemes are discussed below.

1.2 POSITION-BASED VISUAL SERVOING

The position-based visual servo (abbreviated **PBVS**) control scheme is a 3D approach to visual servoing [1] in which the vision sensor is considered as a 3D sensor [2]. In fact, the PBVS approach is just an extension of IBVS approach with the main difference being the number of cameras: PBVS is in fact IBVS with stereovision system [2]. In the position-based control approaches, the vision system is used as a recognition observer to measure the relative positions X_{rel} between the system under consideration and some object in its environment [3].

Consider, for instance, a task where a robot manipulator is used to grasp an object. By using the epipolar geometry, one can obtain the 3D grasp points relative to the camera fixed frame, which in turn serve as the set point. The error to be controlled is obtained by comparing the desired position of the grasp points with the current position. Weiss *et. el.* reported the sequence of operation of a position-based static *'look and move'* system in [3] which is used most often in present industrial applications. The first step is to look at the object and estimate the relative end effector position. In the second step, the difference between the current position and the reference position is computed and a command is issued to move the end effector. The end-effector motion takes place at the last step; the first step is not executed until the robot completes its motion.

Corke in [4] reports two classes of techniques that can be used to estimate the depth of the feature in PBVS approach. One class is called *'active range sensors'* in which the reflected energy beam (ultrasonic or optical) is detected to measure the depth. The other class is of *'passive techniques'* in which no extra sensors are required; instead the depth is estimated through observations made under ambient illumination. These techniques are briefly defined below:

- **Photogrammetric techniques** in which the 3D pose is determined using 2D images and some additional data (*for e.g.,* 3D CAD model).
- In **stereo vision techniques**, the locations of feature points obtained from two different viewpoints are matched to resolve the unknown depth.
- **Depth from motion** technique relies on sequential monocular views that are obtained from different viewpoints to estimate the depth.
- If the camera to target distance is fixed (as assumed in some experiments), then the closed-loop system dynamics can be used to estimate the depth because the closed-loop response is a function of the distance between the camera and the target. Such a technique is called **depth from dynamics.**

The main advantage of PBVS systems is the simplicity of control design. The main problem of PBVS scheme is its lack of robustness due to the fact that it depends primarily on pose estimation which is highly sensitive to camera calibration. If low-cost (consequently low-quality) cameras are used then PBVS approach may be rendered useless to some extent. Guenard *et. el.* in [6] also reported that the PBVS approach has a tendency of image features leaving the camera field of view during operations. IBVS technique can overcome these problems, and is discussed below.

1.3 IMAGE-BASED VISUAL SERVOING

Image-based visual servo scheme, abbreviated as **IBVS**, is a 2D approach to visual servoing in which the motion of robot is controlled by controlling the features in the image plane. The IBVS approach is more robust than the PBVS approach due to the fact that IBVS does not require 3D reconstruction step like the way PBVS does. This kind of approach in which feature feedback is used was first proposed by Sanderson and Weiss in their work [5] published in 1982. In their work, they defined reference and feedback signals in terms of the image feature values corresponding to the current and desired robot positions. The difference between the values of the image features was calculated at every sampling instant and then used by the control law to drive the robot in the desired direction to minimize the error [3].

The goal of traditional IBVS schemes is to minimize the error of the form

$$\delta(t) = p(d(t), a) - p^* \tag{1.1}$$

The vector $d(t)$ is the location of the centroid of an object, and a is a set of parameters representing additional knowledge of the system. Together, they are used to calculate a vector $p(d(t), a)$ of visual features that are a function of the relative pose of the camera to the target. The movement of camera from one point to another will most likely result in movement of two or more features at the same time, therefore, making the function nonlinear and cross-coupled. To overcome this problem, a new approach was introduced by Espiau, Chaumette, and Rives in their 1992 work [7] in which they attempted to linearize the function defining the movement of image features about the operating point using a matrix called **Jacobian matrix**. The *Jacobian matrix*, also known as *feature Jacobian*, *image Jacobian*, *feature sensitivity matrix*, or *interaction matrix* is a matrix that relates the rate of change in pose of the object to the rate of change in feature space.

Generally in IBVS schemes, the depth of each feature of interest appears in the coefficients of interaction matrix related to the translational motion[11]. The estimation of this unknown depth is the most daunting problem of the IBVS approach due to the fact that the depth coordinate is lost when a 3D environment is projected on a 2D image surface. Various approaches have been proposed in the literature to estimate the unknown depth; these include: estimation via partial pose estimation, adaptive control, and estimation of image Jacobian using quasi-Newton techniques. Two newer, yet somewhat similar techniques have also been developed recently that rely on using a different coordinate system than the Cartesian coordinates. The first one was proposed by a team of Hosei University in Tokyo, Japan in their 2005 work [10] in which they used cylindrical coordinate system to resolve the unknown depth. The other approach was proposed by Hamel and Mahony in their 2002 work [8] in which they used spherical coordinates to accurately control the position of UAV motion in three dimensions. This approach was selected for the experimental work documented in here.

1.4 OVERVIEW OF UNMANNED AIRCRAFT AND EXISTING IBVS CONTROLLERS

A vehicle that uses support of air to fly is called an aircraft. Modern day aircraft come in a variety of shapes, sizes, capabilities, and categories to serve various purposes and missions. Different types of aircraft can be classified upon the method of lift, method of propulsion, size, and the domain of applications.

If the classification of method of lift is used then most aircraft will fall in two categories: aerostats or aerodynes. **Aerostats** are aircraft that use the principle of buoyancy to float in air as they are lighter than air. Such aircraft have very limited or no use in the modern era as their direction is very difficult

to control. Examples of such aircraft are hot air balloons or kites. On the other hand, **aerodynes** are aircraft that are heavier than air and are driven by the reaction force that is obtained by pushing the air downwards. *Fixed-wing aircraft* and *rotorcrafts* are the two major categories of aerodyne type aircraft. Helicopters fall in the category of rotorcrafts as they use spinning rotor along with aerofoil shaped rotorwings called blades.

Powered and unpowered aircraft are the two categories of aircraft if the aircraft are to be classified by the means of propulsion. Helicopters fall in the category of powered aircraft as their rotors are driven by using combustion engines, jet turbines, etc. A special arrangement of propellers, in which the propellers are enclosed by a shroud is called **ducted-fan** arrangement. This kind of arrangement is popular for small-scale airborne vehicles as they provide several advantages. Examples of such advantages are safety of operations, increased thrust efficiency, and noise reduction. The primary disadvantages of the ducted-fan arrangement stems from the fact that addition of duct increases weight and cost of the aircraft. Apart from that, the other primary disadvantage of ducted-fan arrangement is that at high speed the drag produced by the duct could be greater than the thrust itself, thereby nullifying the advantage of adding a duct. As far as the scope of this experimental work is concerned, the aircraft would be flying at relatively low speed and only in a laboratory environment, and therefore, a ducted-fan arrangement would be more suitable than a conventional arrangement and is therefore chosen. The choice of using this kind of arrangement can further be justified by an analysis of popularity of use of this kind of aircraft structure by different research groups with [28],[29], and [30] being a few out of numerous groups using ducted fan configuration for experimental purposes.

Helicopters are a class of *Vertical Take-Off and Landing* (VTOL) vehicles. The idea of rotary wing flight is a centuries old idea with earliest record being of Leonardo da Vinci attempting to engineer a vehicle capable of moving up and forward which however did not materialize. As the understanding of rigid-body flight dynamics improved over time, scientists and engineers were able to eventually design a rotorcraft plane capable of generating and maintaining lift over extended periods of time by early twentieth century. These aircraft were manned and with the passage of time along with advances in technology, the category of manned aircraft evolved into unmanned aircraft. Unmanned helicopters can therefore be classified as **VTOL UAVs**.

VTOL UAVs have been around for over half a century but their true potential has only been realized in the recent past [36]. As such a lot of literature on dynamics analysis of helicopters and helicopter design can be easily found, for ex. [31], [32], and in [37] but the literature on control systems for UAVs is scarce. Even though, the work on vision-based autonomous helicopter project has been underway at Carnegie-Mellon research institute since 1991 [24], relatively few advances have been made in that area and vision-based control of UAVs is still

an area of control in its infancy. Most of the published work pertaining to vision-based control of UAVs is done by the team of Hamel and Mahony. In [6], the team experimented the theoretical controller proposed in [8]. In [33], experimental results are presented of a control law capable of controlling hovering flight and landing manoeuvre of UAV using optical flow. In [35], a coupled filter algorithm is presented that can estimate the pose and velocity of a UAV using vision sensors along with accelerometers and gyroscopes. This work was further extended in [34], and an IBVS controller was proposed that does not require direct measurement of linear velocity. In all the works mentioned above, the common assumption is that the target is stationary.

Some potential applications of vision-based autonomous helicopter include crime-fighting or cinematography applications. In such applications, the target would most likely be non-stationary, i.e. it will be moving. While plenty of literature can be found on IBVS control of stationary targets, there is hardly any work done on tracking a target that is moving. In some cases, such as done in [9], the assumptions made to develop an IBVS controller make the task over simplified which in turn make the practicality of the controller somewhat ambiguous.

The overall goal of the experimental work documented here is to develop an IBVS controller to track a target moving at a constant unknown velocity. The IBVS controller controls the position and velocity of the UAV in all three coordinates, i.e., x, y, and z. The UAV is assumed to be hovering at some certain distance away from the moving target. When the IBVS controller is initialized, it takes the UAV to the target and then the UAV hovers on top of the target along its path. This way, both the position error and the velocity error are minimized at the same time.

Last, but not least, some of the weaknesses of IBVS controllers should also be mentioned. One key problem of IBVS scheme besides the depth problem mentioned above is that the rigid-body dynamics of the camera ego-motion are highly coupled when expressed as target motion in the image plane [8]. It is also reported by Chaumette and Hutchinson in [2] that only local asymptotic stability can be obtained for IBVS controllers. There are also major difficulties associated with real time implementation of IBVS control algorithms. These difficulties include sampling rate, processing of only the information of desired features instead of the whole image, tracking of features along the sequence, etc. These problems are beyond the scope of this work but should be properly dealt with to achieve proper and successful results of the IBVS control algorithm.

Chapter 2

SYSTEM MODELING AND DESIGN

2.1 SYSTEM MODELING

System modeling is vital for any kind of design and analysis in the field of applied sciences. It is a skeletal model of the system which is used to express, visualize, and analyze the architecture of the system. It assists in development of large and complex systems by reducing development costs and time, allowing for simple and easy integration of ready made components, and letting the developer visualize the weaknesses of the system.

The system proposed for experiments is called **Ducted Fan Type VTOL UAV**. The system design using CAD tools will be discussed in the next section. The model governing the dynamics of UAVs is derived from the application of fundamental principles of mechanics. The rotational dynamics of a rigid-body evolving in air can be expressed in different forms depending upon the type of attitude representation. The most commonly used types of attitude representations are Euler angles, rotation matrices, and unit quaternion. For more information, review, or analysis of different types of attitude representation, interested reader may refer to [12],[13], and [14]. By using variation of momentum principle, as done in [15], different system models can be derived for a rigid-body evolving in air, a good review of which is presented in [16]. For a typical spacecraft evolving in air, its motion in the body-fixed frame $\mathcal{B}$ is given by (refer to [17] for derivation).

$$\dot{\xi} = R^T V \tag{2.1}$$

$$\dot{V} = -S(\Omega)V + \mu - S(\frac{\hat{z}}{ml})\tau_a + \frac{1}{m}F_d \tag{2.2}$$

$$\dot{R} = -S(\Omega)R \tag{2.3}$$

$$I_b\dot{\Omega} = -S(\Omega)I_b\Omega + \epsilon S(\hat{z})F_d + \tau_a \tag{2.4}$$

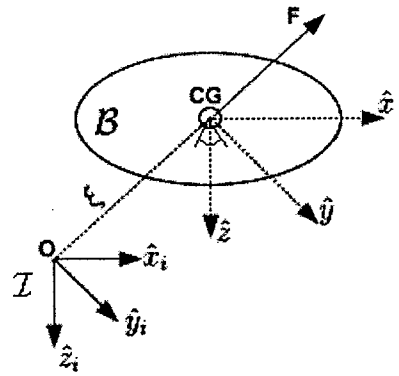

Figure 2.1: The $\mathcal{I}$ and $\mathcal{B}$ reference frames

where, μ denotes the forces acting on the system and is given by the expression $\mu = gR\hat{z} - \frac{T}{m}\hat{z}$.

In the above expressions, the first two equations govern the translational dynamics while the last two govern the rotational dynamics of a spacecraft. Suppose $\mathcal{I} = [\hat{x}_i, \hat{y}_i, \hat{z}_i]^T$ denotes a right-hand inertial frame such that $\hat{z}_i$ is pointing downwards into the earth. Let $\mathcal{B}$ denote the right-hand body fixed frame given by $\mathcal{B} = [\hat{x}, \hat{y}, \hat{z}]^T$, the positive $\hat{z}$ direction points downwards. It should be noted that the position of the camera, given by $\xi = [x, y, z]^T \in \mathcal{I}$ is measured at the center of camera instead of center of mass of the vehicle. The description of the other symbols is as follows:

$\boldsymbol{R}$: *Rotation matrix of frame $\mathcal{I}$ w.r.t. frame $\mathcal{B}$.*
$\boldsymbol{V}$: *Velocity of the aircraft in $\mathcal{B}$*
$\boldsymbol{m}$: *Mass of the aircraft.*
τ_a: *Control input.*
F_d: *External disturbance force.*
Ω: *Angular velocity of the spacecraft expressed in $\mathcal{B}$.*
ϵ: *Distance from the center of $\mathcal{B}$ to the point where the disturbance forces are applied.*
$\boldsymbol{l}$: *Horizontal distance from the $\hat{z}$ axis to the center of the aileron.*
I_b: *Inertia matrix in $\mathcal{B}$.*

The notation $S(\Omega)$ denotes the skew-symmetric matrix of Ω such that $S(\Omega)$V $= \Omega$ x V (i.e. the vector cross-product between two vectors Ω and V. For a vector $\Omega = [\Omega_x, \Omega_y, \Omega_z]$, $S(\Omega)$ is given by:

$$S(\Omega) = \begin{bmatrix} 0 & -\Omega z & \Omega y \\ \Omega z & 0 & -\Omega x \\ -\Omega y & \Omega x & 0 \end{bmatrix}$$

The UAV is designed in such a way that it is symmetric with respect to planes $(\hat{x}, \hat{y})$ and $(\hat{y}, \hat{z})$. The center of gravity **G** of the vehicle therefore resides at the $\hat{z}$ axis, which is also the principle axis of symmetry. This fact helps in the simplification of the inertia matrix, which can be written in the simplified form as:

$$I_b = \begin{bmatrix} I_1 & 0 & 0 \\ 0 & I_1 & 0 \\ 0 & 0 & I_2 \end{bmatrix} \tag{2.5}$$

In the dynamical model given above in Eq. (2.1) - (2.4), the control input τ_a appears in both the translational dynamics as well as in the rotational dynamics equations. The appearance of control input in both expressions make the UAV control problem more challenging. To overcome this problem, a simple technique of change of coordinates was introduced by Olfati-Saber in his work [27]. By extending the work from 2D, as proposed in the original work [27], to 3D as done in [28], the control input can easily be removed from translational dynamics relations. Consider the following change of variables,

$$\overline{V} = V + S(\frac{\hat{z}}{ml}) I_b \Omega \tag{2.6}$$

The translational dynamics become

$$\dot{\xi} = R^T \left(\overline{V} - S\left(\frac{\hat{z}}{ml}\right) I_b \Omega \right)$$

$$\dot{\overline{V}} = -S(\Omega)\overline{V} + \mu + (S(\Omega)S(\frac{\hat{z}}{ml}) - S(\frac{\hat{z}}{ml})S(\Omega)) I_b \Omega + f(\epsilon, m, l, F_d)$$

Using the identity

$$\left(S(\Omega)S(\frac{\hat{z}}{ml}) - S(\frac{\hat{z}}{ml})S(\Omega) \right) I_b \Omega = \frac{1}{ml} S(I_b \Omega) S(\Omega) \hat{z}$$

the translational dynamics can be rewritten as

$$\dot{\xi} = R^T(\overline{V} - S(\frac{\hat{z}}{ml})I_b\Omega)$$

$$\dot{\overline{V}} = -S(\Omega)\overline{V} + \mu - \frac{1}{ml}S(I_b\Omega)S(\Omega)\hat{z} + f(\epsilon, m, l, F_d)$$

where $f(\epsilon, m, l, F_d) = \frac{1}{ml}(lI_{3\times 3} + \epsilon S^2(\hat{z}))F_d = [\frac{(l-\epsilon)F_{d1}}{ml}, \frac{(l-\epsilon)F_{d2}}{ml}, \frac{F_{d3}}{ml}]^T$, with $F_d = [F_{d1}, F_{d2}, F_{d3}]^T$. By using the fact that I_b =diag$[I_1, I_1, I_2]$, and letting $\overline{\xi} = \xi - \frac{I_1}{ml}R^T\hat{z}$, it can be shown that

$$\overline{v} = v - \frac{I_1}{ml}R^T S(\Omega)\hat{z}$$

where $v = \dot{\xi}$ and $\overline{v} = \dot{\overline{\xi}}$. The velocity $\overline{v}$ can be expressed in the body-fixed frame as

$$\begin{aligned} R\overline{v} &= Rv - \frac{I_1}{ml}RR^T S(\Omega)\hat{z} \\ \overline{V} &= V + \frac{I_1}{ml}S(\hat{z})\Omega \\ &= V + S(\frac{\hat{z}}{ml})I_b\Omega \end{aligned}$$

where we used the fact that $\frac{I_1}{ml}S(\hat{z}) = S(\frac{\hat{z}}{ml})I_b$ and $S(\Omega)\hat{z} = -S(\hat{z})\Omega$. The symbols v and $\overline{v}$ have usual meanings as above but are used to express the respective velocities in $\mathcal{I}$. Let $\overline{\xi}_b = R\overline{\xi}$ denote the position of the aircraft in the body-fixed frame. Hence the full dynamical model of the system is given by the following four equations.

$$\dot{\overline{\xi}}_b = -S(\Omega)\overline{\xi}_b + \overline{V} \tag{2.7}$$

$$\dot{\overline{V}} = -S(\Omega)\overline{V} + \mu - \frac{1}{ml}S(I_b\Omega)S(\Omega)\hat{z} + f(\epsilon, m, l, F_d) \tag{2.8}$$

$$\dot{R} = -S(\Omega)R \tag{2.9}$$

$$I_b\dot{\Omega} = -S(\Omega)I_b\Omega + \epsilon S(\hat{z})F_d + \tau_a \tag{2.10}$$

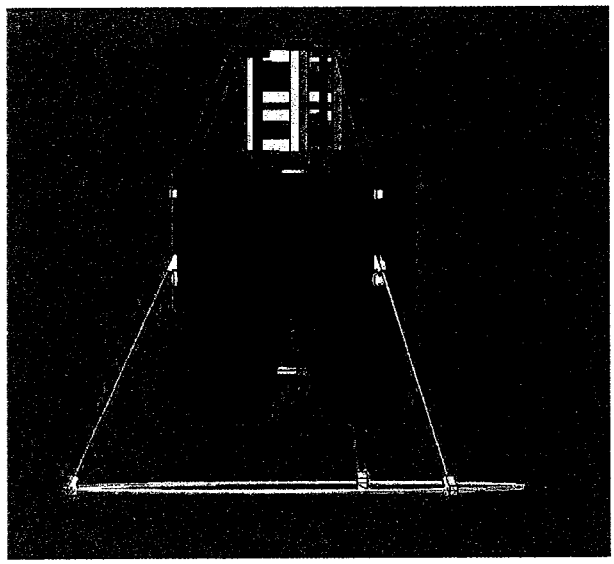

Figure 2.2: The complete UAV model.

2.2 SYSTEM DESIGN

The system, as shown in Fig.2.2, can be crudely partitioned in to three distinct sections. The top section is called the 'nose cone' and is designed to house most of the avionics components, the camera, along with a battery that powers the propeller motors. The duct encloses the two counter rotating propellers that are attached to two separate motors. Each motor is attached to an aluminum plate that is connected to the duct using a link (carbon fibre tubes or triangles). At the exit of the duct, one can find four sets of ailerons with each set comprising of three individual ailerons connected to each other using plastic shafts. One end of the ailerons is used to attach an aluminum plate that holds the tail cone. The 'tail cone' is the structure found below the duct and holds a battery, sonar, and few other electronic components. At the very bottom of the system is the landing ring attached to the duct using carbon fibre tubes and plastic brackets. The particulars of various components can be found in Appendix 'A', and the overall system architecture is described in detail in the Chapter entitled "Experimental Setup".

Chapter 3

ERROR DYNAMICS MODELING

In this chapter, the derivation of the error function is presented. It is important to understand the need to use spherical projection over perspective projection before the derivation of the model.

3.1 SPHERICAL PROJECTION

Let $\overline{P}_i \in \mathcal{I}$ for i = 1,...,n denote the unknown coordinates of target points moving at a constant velocity with respect to the body-fixed frame $\mathcal{B}$ and $\overline{\xi}_b$ be the position of camera in $\mathcal{B}$. Let $P_i = [X_i, Y_i, Z_i]^T$ denote the coordinates of $\overline{P}_i$ in the body-fixed frame $\mathcal{B}$. Let p_i denote the projection of the point P_i on the image surface. For any camera surface, the observed image feature p_i of the point P_i is given by

$$p_i = \frac{1}{r(P_i)} P_i \tag{3.1}$$

where $r(P_i)$ is the relative depth of the target. When a 3D scene is projected on a flat 2D image surface, the size of the objects in the environment is scaled down depending upon the depth, which also results in loss of depth coordinates on the image plane. The loss of the depth axis for perspective projection can be mathematically modeled by

$$p_i = \frac{f}{Z_i} P_i = \begin{bmatrix} f\frac{X_i}{Z_i} \\ f\frac{Y_i}{Z_i} \\ f \end{bmatrix} \tag{3.2}$$

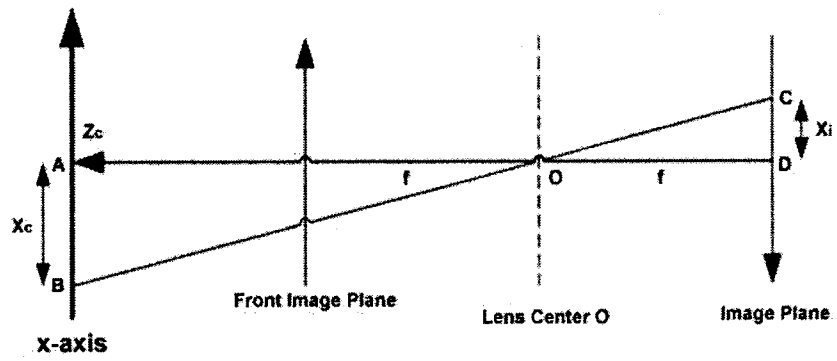

Figure 3.1: The perspective projection model. Figure adapted from [18]

where f is the focal length of the camera. The perspective imaging model is shown pictorially in Fig.3.1

The loss of depth coordinate poses significant constraints in the practical 3D applications of visual servo algorithms. To overcome this problem, several techniques were developed by various different research teams. The approach of spherical projection was used by Hamel and Mahony in [25] to successfully control the motion of UAV in 3D using only one camera. By using spherical projection, it is possible to obtain some information about the depth coordinate in the form of a variable changing with respect to the other two position coordinates. For a spherical surface, the scaling function $r(P_i)$ is given by

$$r(P_i) = \frac{|P_i|}{f}$$

where $|P_i|$ denotes the Euclidean norm of the point P_i. Therefore, the expression of the projection of the point P_i is

$$p_i = \frac{f}{Z_i} P_i = \begin{bmatrix} \frac{fX_i}{Z_i\sqrt{X_i^2+Y_i^2+Z_i^2}} \\ \frac{fY_i}{Z_i\sqrt{X_i^2+Y_i^2+Z_i^2}} \\ \frac{f}{\sqrt{X_i^2+Y_i^2+Z_i^2}} \end{bmatrix}$$

It is important to note that it is not necessary to implement a spherical camera to obtain spherical projection. The perspective projection from an or-

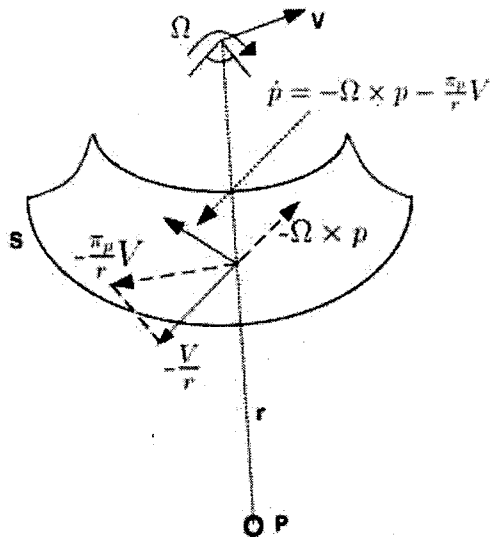

Figure 3.2: The dynamics of a point moving on a sphere. Figure adapted from [8]

dinary camera with flat image plane can be utilized to calculate the spherical projection, i.e. $p_i = \frac{p_{p_i}}{|p_{p_i}|}$ where p_{p_i} denotes the perspective projection of the point P_i. In order to calculate the dynamics of a point projected on a sphere, the derivative of $r(P_i)$ will be required in later stages and is given by

$$\frac{d}{dt} r(P_i) = \eta_{P_i}^T (\dot{P}_i) \tag{3.3}$$

where $\eta_{P_i} = \frac{\partial}{\partial \tau} r(P_i)$ denotes the differential of the scaling function $r(P_i)$.

3.2 ERROR DYNAMICS MODELING

Let,

$$P_i = R(\overline{P}_i) - \overline{\xi}_b \,. \tag{3.4}$$

denote the distance between the camera and the target expressed in $\mathcal{B}$. The time derivative of $\dot{P}_i$ is

$$\begin{aligned}\dot{P}_i &= \dot{R}\overline{P}_i - R\dot{\overline{P}}_i - \dot{\overline{\xi}}_b \\ &= -S(\Omega)R\overline{P}_i + R\dot{\overline{P}}_i + S(\Omega)\overline{\xi}_b - \overline{V} \\ &= -S(\Omega)P_i + V_T - \overline{V} \end{aligned} \tag{3.5}$$

where we used that fact that $R\dot{\overline{P}}_i = V_T$. The term V_T denotes the target velocity in $\mathcal{B}$. As mentioned earlier, the spherical projection of the point P_i is given by

$$p_i = \frac{1}{r(P_i)} P_i$$

and the derivative of p_i is obtained using chain rule for derivatives.

$$\dot{p}_i = \frac{1}{r(P_i)}\dot{P}_i - \frac{\frac{d}{dt}r(P_i)}{r(P_i)^2}P_i$$

By substituting the dynamics of the point P_i from Eq. 3.5, one obtains

$$\begin{aligned}\dot{p}_i &= \frac{1}{r(P_i)}[-S(\Omega)P_i + V_T - \overline{V}] - \frac{1}{r(P_i)^2}P_i\eta_{p_i}^T[-S(\Omega)P_i + V_T - \overline{V}] \\ &= -S(\Omega)p_i + \frac{V_T - \overline{V}}{r(P_i)} - p_i\eta_{p_i}^T[-S(\Omega)]p_i + p_i\eta_{p_i}^T\frac{V_T - \overline{V}}{r(P_i)}\end{aligned}$$

As explained in [8], in order to obtain passive terms in the final image dynamics expression (i.e., terms that cancel out or do not play role in affecting the stability of the system), it is important that the first two terms should not contribute to the derivative of Lyapunov function (i.e., $-p_i^T S(\Omega)p_i + p_i^T p_i \eta_{p_i}^T S(\Omega)p_i = 0$). Due to the fact that $S(\Omega)$ is skew symmetric, the first term will automatically disappear but the second term will not unless $\eta_{p_i}^T$ is set to p_i^T. Therefore, by choosing $\eta_{p_i} = \lambda(P_i)p_i$ where $\lambda(P_i)$ is some scalar function, it is possible to obtain the desired passivity-like properties in error dynamics equation [8]. The dynamics of the point $\dot{p}_i$ can be written in a simplified form as

$$\dot{p}_i = -S(\Omega)p_i + (I_{3\times 3} - p_i p_i^T)\frac{V_T - \overline{V}}{r(P_i)} \tag{3.6}$$

Assume that the desired features are moving at the same velocity as the target itself. The desired image features P_i^* are given in the body fixed frame by the expression

$$P_i^* = R\overline{P}_i^* - \overline{\xi}_b^*$$

the dynamics of which are

$$\begin{aligned} \dot{P}_i^* &= \dot{R}\overline{P}_i^* + R\dot{\overline{P}}_i^* - \dot{\overline{\xi}}_b^* \\ &= -S(\Omega)R\overline{P}_i^* + R\dot{\overline{P}}_i^* + S(\Omega)\overline{\xi}_b^* - \overline{V}^* \\ &= -S(\Omega)[R\overline{P}_i^* - \overline{\xi}_B^*] + R\dot{\overline{P}}_i^* - \overline{V}^* \end{aligned}$$

Assume that the desired image features move on the image surface due to the ego-motion of the camera only and not due to the motion of the target (The term "ego-motion" refers to self-motion of the camera as it moves with the motion of UAV). With this assumption, the last two terms in the above equation will cancel out, and the simplified dynamics of the desired features can be written as

$$\dot{P}_i^* = -S(\Omega)P_i^* \tag{3.7}$$

The spherical projection of the point P_i^* is given by

$$p_i^* = \frac{1}{r(P_i^*)}P_i^* \tag{3.8}$$

the dynamics of which are

$$\dot{p}_i^* = \frac{1}{r(P_i^*)}\dot{P}_i^* + \frac{\frac{d}{dt}r(P_i^*)}{r(P_i^*)^2}P_i^*$$

By using Eq. 3.7, the above dynamical expression can be written as

$$\begin{aligned}\dot{p}_i^* &= \frac{-S(\Omega)}{r(P_i^*)}P_i^* + \frac{P_i^*\eta^{*T}[-S(\Omega)P_i^*]}{r(P_i^*)^2} \\ &= -S(\Omega)p_i^* - p_i^*\eta^{*T}S(\Omega)p_i^*\end{aligned}$$

Choose $\eta^* = \lambda(p_i^*)p_i^*$ to obtain the desired passivity like properties in the image dynamics. Therefore, the desired image features can be written in simplified form as

$$\dot{p}_i^* = -S(\Omega)p_i^* \tag{3.9}$$

The error is given as the difference between the vectors p_i and p_i^* which can be written as

$$\delta_i = p_i - p_i^*$$

the derivative of which is

$$\begin{aligned}\dot{\delta}_i &= -S(\Omega)p_i + (I_{3\times 3} - p_i p_i^T)\frac{V_T - \overline{V}}{r(P_i)} + S(\Omega)p_i^* \\ &= -S(\Omega)(p_i - p_i^*) + (I_{3\times 3} - p_i p_i^T)\frac{V_T - \overline{V}}{r(P_i)} \\ &= -S(\Omega)\delta_i + (I_{3\times 3} - p_i p_i^T)\frac{V_T - \overline{V}}{r(P_i)} \\ &= -S(\Omega)\delta_i + (I_{3\times 3} - p_i p_i^T)\frac{V_T - \overline{V}}{r(P_i)}\end{aligned}$$

Note that the term $\frac{I_{3\times 3} - p_i p_i^T}{r(P_i)}$ is a positive semi-definite term, which complicates the control design. To overcome this problem, the number of target points should be greater than or equal to two as shown in [8]. To this end, we consider

the centroids of p_i and p_i^* as follows:

$$\begin{aligned} p &:= \sum_{i=1}^{n} p_i \\ p^* &:= \sum_{i=1}^{n} p_i^* \\ \delta_p &:= \sum_{i=1}^{n} \delta_i \\ &= \sum_{i=1}^{n} p_i - \sum_{i=1}^{n} p_i^* \\ Q &:= \sum_{i=1}^{n} \frac{I_{3\times 3} - p_i p_i^T}{r(P_i)} \end{aligned}$$

Using the above variables, the full tracking error dynamics of the error δ_p can be written as

$$\dot{\delta_p} = -S(\Omega)\delta_p + Q[V_T - \overline{V}] \tag{3.10}$$

$$\dot{\overline{V}} = -S(\Omega)\overline{V} + \mu - \frac{1}{ml}S(I_b\Omega)S(\Omega)\hat{z} + f(\epsilon, m, l, F_d) \tag{3.11}$$

$$\dot{R} = -S(\Omega)R \tag{3.12}$$

$$I_b\dot{\Omega} = -S(\Omega)I_b\Omega + \epsilon S(\hat{z})F_d + \tau_a \tag{3.13}$$

The above error dynamics will be used in the next chapter to derive the control law.

Chapter 4

CONTROL DESIGN

In this chapter, two IBVS control laws capable of tracking a moving target are derived. For the first control design, it is assumed that the target velocity is known while the second control design will be for a target moving at a constant unknown velocity. It is assumed that the UAV is hovering in air at some distance away from the target and the target is moving in a small neighborhood.

4.1 CONTROL DESIGN 1

As mentioned above, for the first control design the target velocity V_T is assumed to be known. In the error dynamics given in Eq. 3.10, there are only two variables that can be used for virtual control. One is Ω and the other one is $\overline{V}$. $\overline{V}$ is a better choice to be used as a virtual control over Ω because the first order dynamics of Ω makes the control design complicated. In order to derive the controller, let us use $\overline{V}$ as virtual control and set it as

$$\overline{V} = V_T + k_1 \delta_p \tag{4.1}$$

where k_1 is a scalar gain. For a Lyapunov function candidate of the form $\frac{1}{2}\delta_p^T \delta_p$, using the virtual control of $\overline{V}$ as mentioned above would result in the derivative of Lyapunov function to be negative definite in δ_p and would therefore guarantee the exponential convergence of the error δ_p.

The velocity $\overline{V}$ can not be directly assigned in a physical system due to the fact that τ_a is the control input. To continue the back-stepping procedure, let δ_v, the **velocity error**, be the error between the camera velocity and the target velocity plus an additional term δ_p, i.e.

$$\delta_v = \overline{V} - k_1\delta_p - V_T \tag{4.2}$$

With the above choice of δ_v, the time derivative of δ_p will be given by Eq. 4.3 shown below

$$\dot{\delta}_p = -S(\Omega)\delta_p + Q\left(V_T - \overline{V}\right) - Q\delta_2 \tag{4.3}$$

The dynamics of the error δ_v in view of Eq. 3.10 and Eq. 3.11 will be given

$$\begin{aligned}\dot{\delta_v} &= \dot{\overline{V}} - k_1\dot{\delta_p} - \dot{V}_T \\ &= \mu - \frac{1}{ml}S(I_b\Omega)S(\Omega)\hat{z} + f(\epsilon, m, l, F_d) + k_1Q\delta_v - k_1Q(-k_1\delta_p) \\ &\quad -S(\Omega)V_T - \dot{V}_T\end{aligned} \tag{4.4}$$

Let,

$$\mathcal{L} = \frac{1}{2}\delta_p^T\delta_p + \frac{1}{2}\delta_v^T\delta_v + \frac{1}{2\Gamma}\tilde{F}^T\tilde{F} \tag{4.5}$$

be a candidate Lyapunov function where $\tilde{F} = F_d - \hat{F}_d$ and Γ is a positive scalar gain. The time derivative of the Lyapunov function in view of Eq. 3.10 and Eq. 4.4 is given by

$$\begin{aligned}\dot{\mathcal{L}} &= \delta_p^T\left(Q(-k_1\delta_p) - Q\delta_v\right) + \delta_v^T\left(\mu - \frac{1}{ml}S(I_b\Omega)S(\Omega)\hat{z} + \frac{1}{m}(lI_{3\times3} + \epsilon S^2(\hat{z}))F_d\right. \\ &\quad \left. +k_1Q\delta_v - S(\Omega)V_T - k_1Q(-k_1\delta_p) - \dot{V}_T\right) - \frac{1}{\Gamma}\tilde{F}^T\dot{\hat{F}}_d\end{aligned}$$

Set the control effort μ as

$$\mu = \frac{1}{ml}S(I_b\Omega)S(\Omega)\hat{z} - \frac{1}{ml}(lI_{3\times3} + \epsilon S^2(\hat{z}))\hat{F}_d + S(\Omega)V_T - k_2\delta_v + \dot{V}_T \tag{4.6}$$

where k_2 is a scalar gain satisfying the condition $k_2 > k_1\lambda_{max}(Q)$. The derivative of the Lyapunov candidate function can be simplified as

$$\begin{aligned}\dot{\mathcal{L}} &= \delta_p^T\left(Q(-k_1\delta_p) - Q\delta_v\right) + \delta_v^T\left(k_1Q\delta_v + k_1Q\delta_p - k_2\delta_v + \frac{1}{ml}(lI_{3\times3} + \epsilon S^2(\hat{z}))\tilde{F}\right) \\ &\quad -\frac{1}{\Gamma}\tilde{F}^T\dot{\hat{F}}_d\end{aligned}$$

Let,

$$\dot{\hat{F}}_d = \frac{\Gamma}{ml}(lI_{3\times3} + \epsilon S^2(\hat{z}))\delta_v \tag{4.7}$$

be the derivative of the estimation of the unknown disturbance force. Finally, the derivative of Lyapunov candidate function can be written as

$$\begin{aligned}\dot{\mathcal{L}} &= -k_1\delta_p^TQ\delta_p - (1-k_1)\delta_p^TQ\delta_v - \delta_v^T\left[k_2I_{3\times3} - k_1Q\right]\delta_v &(4.8)\\ &= -\begin{bmatrix}\delta_p^T & \delta_v^T\end{bmatrix}\begin{bmatrix}k_1Q & \frac{1-k_1}{2}Q \\ \frac{1-k_1}{2}Q & k_2I_{3\times3} - k_1Q\end{bmatrix}\begin{bmatrix}\delta_p \\ \delta_v\end{bmatrix} &(4.9)\end{aligned}$$

In order to guarantee convergence of the errors δ_p and δ_v, the Young's inequality can be applied to obtain conditions on gains k_1 and k_2. Using Young's inequality, one obtains the condition

$$\begin{aligned}\dot{\mathcal{L}} &\leq -k_1\lambda_{min}(Q)||\delta_p||^2 - (k_2 - k_1\lambda_{max}(Q))||\delta_v||^2 \\ &\quad +(1+k_1)\left(\epsilon_1||\delta_p||^2 + \frac{1}{4\epsilon_1}\lambda_{max}^2(Q)||\delta_v||^2\right) \\ &\leq -[k_1\lambda_{min}(Q) - \epsilon_1(1+k_1)]||\delta_p||^2 - [k_2 - k_1\lambda_{max}(Q) - \frac{k_1+1}{4\epsilon_1}\lambda_{max}^2(Q)]||\delta_v||^2\end{aligned}$$

From the above expressions, it can be concluded that the following two conditions must be satisfied

$$\begin{aligned}k_1\lambda_{min}(Q) - \epsilon_1(1+k_1) &> 0 \\ k_2 - k_1\lambda_{max}(Q) - \frac{k_1+1}{4\epsilon_1}\lambda_{max}^2(Q) &> 0\end{aligned}$$

for any $\epsilon_1 > 0$, where ϵ_1 is a Young's equation constant. This result guarantees the global boundedness of δ_p and δ_v as well as their convergence to zero when time goes to infinity.

REMARK 4.1: If the gain $k_1 = 1$, then satisfying the condition $k_2 > k_1\lambda_{max}(Q)$ is enough to obtain exponential convergence of the errors δ_p and δ_v to zero. In such a case, there will be no need to select gains using Young's inequality.

4.2 CONTROL DESIGN 2

In this section, an IBVS control law capable of following a moving target with an unknown velocity will be derived. The adaptive term $\hat{V}_T$ will estimate the unknown constant velocity V_T and the error $\tilde{V}$ will be used to measure the difference between $\hat{V}_T$ and V_T. Once again, pick $\bar{V}$ as virtual control and set it as

$$\bar{V} = \hat{V}_T + \bar{k}\delta_p \tag{4.10}$$

and let $\tilde{V}$ be equal to

$$\tilde{V} = \hat{V}_T - V_T - k\delta_p \tag{4.11}$$

where $\bar{k}$ and k are scalar gains greater than zero ($\bar{k} > 0$, $k > 0$). The following development using $\mathcal{L}_1$ as a Lyapunov function candidate shows why the errors δ_p and $\tilde{V}$ can be minimized but not guaranteed to converge to zero. The backstepping procedure to complete the control design is continued from there on. Let,

$$\mathcal{L}_1 = \frac{1}{2}\delta_p^T\delta_p + \frac{1}{2}\tilde{V}^T\tilde{V} \tag{4.12}$$

be a candidate Lyapunov function. Using the dynamics of δ_p from Eq. 3.10 and $\tilde{V}$ from Eq. 4.11, the derivative of $\mathcal{L}_1$ is given by

$$\dot{\mathcal{L}}_1 = \delta_p^T\left(-S(\Omega)\delta_p + Q(V_T - \overline{V})\right) + \tilde{V}^T\left(\dot{\hat{V}}_T + kS(\Omega)\delta_p - kQ(V_T - \overline{V})\right) \tag{4.13}$$

Set $\dot{\hat{V}}_T$ as follows:

$$\dot{\hat{V}}_T = -kS(\Omega)\delta_p - \beta(\hat{V}_T - k\delta_p) \tag{4.14}$$

with β being a scalar gain greater than zero ($\beta > 0$). The derivative of the Lyapunov function candidate $\dot{\mathcal{L}}_1$ will therefore be given by

$$\dot{\mathcal{L}}_1 = \delta_p^T Q(V_T - \hat{V}_T - \overline{k}\delta_p) + \tilde{V}^T\left(-\beta(\tilde{V} + V_T) - kQ(V_T - \hat{V}_T - \overline{k}\delta_p)\right)$$

which can be simplified as

$$\begin{aligned}\dot{\mathcal{L}}_1 &= -(k+\overline{k})\delta_p^T Q\delta_p - \tilde{V}^T\left(\beta I_{3\times 3} - kQ\right)\tilde{V} - \tilde{V}^T\left(Q - (k+\overline{k})kQ\right)\delta_p \\ &\quad -\tilde{V}^T\beta V_T\end{aligned} \tag{4.15}$$

Due to the structure of the derivative Lyapunov function candidate (i.e. square terms in δ_p and $\tilde{V}$, cross term of δ_p and $\tilde{V}$, and an extra V_T term), the exponential convergence of δ_p to zero can not be guaranteed because of the "disturbance" term $\tilde{V}^T\beta V_T$. In order to guarantee the boundedness of the error δ_p, the eigenvalues of term $[\beta I_{3\times 3} - kQ]$ should be greater than zero.

The virtual control of $\overline{V}$ can not be directly assigned and backstepping procedure needs to be continued. Let δ_v denote the difference between the camera velocity and the target velocity, plus an additional term in δ_p. Therefore, δ_v is given by

$$\delta_v = \overline{V} - \hat{V}_T - \overline{k}\delta_p \tag{4.16}$$

The dynamics of the error δ_v are given by

$$
\begin{aligned}
\dot{\delta}_v &= \dot{\overline{V}} - \dot{\hat{V}}_T - \overline{k}\dot{\delta}_1 \\
&= -S(\Omega)\overline{V} + \mu - \frac{1}{ml}S(I_b\Omega)S(\Omega)\hat{z} + f(\epsilon, m, l, F_d) - \\
&\quad \left(-kS(\Omega)\delta_p - \beta(\hat{V}_T - k\delta_p)\right) - \overline{k}\left(-S(\Omega)\delta_p + Q(V_T - \overline{V}) - Q\delta_v\right)
\end{aligned} \tag{4.17}
$$

Let,

$$
\mathcal{L} = \frac{1}{2}\delta_p^T\delta_p + \frac{1}{2}\delta_v^T\delta_v + \frac{1}{2}\tilde{V}^T\tilde{V} + \frac{1}{2\Gamma}\tilde{F}^T\tilde{F} \tag{4.18}
$$

be a Lyapunov candidate function with $\tilde{F} = F_d - \hat{F}_d$ and Γ is a positive scalar gain as before. The time derivative of this Lyapunov candidate function, in view of Eq. 3.10, Eq. 4.11, Eq. 4.14, and Eq. 4.17, is

$$
\begin{aligned}
\dot{\mathcal{L}} &= \delta_p^T\left(Q(V_t - \hat{V}_T - \overline{k}\delta_p) - Q\delta_v\right) + \tilde{V}^T\left(-kS(\Omega)\delta_p - \beta(\hat{V}_T - k\delta_p) - k(-S(\Omega)\delta_p\right. \\
&\quad \left. +Q(V_T - \hat{V}_T - \overline{k}\delta_p) - Q\delta_v\right) + \delta_v^T\left(-S(\Omega)\overline{V} + \mu - \frac{1}{ml}S(I_b\Omega)S(\Omega)\hat{z}\right. \\
&\quad \left. +f(\epsilon, m, l, F_d) + kS(\Omega)\delta_p + \beta(\hat{V}_T - k\delta_p) + \overline{k}S(\Omega)\delta_p + \overline{k}Q\delta_v - \right. \\
&\quad \left. \overline{k}Q(V_T - \hat{V}_T - \overline{k}\delta_p)\right) - \tilde{F}^T\dot{\hat{F}}_d
\end{aligned}
$$

Set the control effort μ as

$$
\begin{aligned}
\mu &= S(\Omega)\overline{V} + \frac{1}{ml}S(I_b\Omega)S(\Omega)\hat{z} - (k + \overline{k})S(\Omega)\delta_p - \beta(\hat{V}_T - k\delta_p) - \\
&\quad \frac{1}{ml}(lI_{3\times 3} + \epsilon S^2\hat{z})\hat{F}_d - k_2\delta_v
\end{aligned} \tag{4.19}
$$

with k_2 being a scalar gain greater than the maximum eigenvalue of the matrix Q (i.e. $k_2 > \lambda_{max}(\mathrm{Q})$). Let

$$
\dot{\hat{F}}_d = \frac{\Gamma}{ml}(lI_{3\times 3} + \epsilon S^2(\hat{z}))\delta_v \tag{4.20}
$$

be the derivative of the estimation of the unknown disturbance force. The derivative of the Lyapunov function candidate can be written as

$$\begin{aligned}
\dot{\mathcal{L}} &= \delta_p^T \left(Q(-\tilde{V} - (k+\overline{k})\delta_p) - Q\delta_v \right) + \delta_v^T \left(\overline{k}Q\delta_v - \overline{k}Q(-\tilde{V} - (k+\overline{k})\delta_p) \right. \\
&\quad \left. -k_2\delta_v \right) + \tilde{V}^T \left(-\beta(\hat{V}_T - k\delta_p) - kQ(-\tilde{V} - (k+\overline{k})\delta_p) + kQ\delta_v \right) \\
&= -(k+\overline{k})\delta_p^T Q\delta_p - \delta_v^T \left(-\overline{k}Q + k_2 I_{3\times 3} \right) \delta_v - \tilde{V}^T(\beta I_{3\times 3} - kQ)\tilde{V} \\
&\quad - \left(1 - \overline{k}(k+\overline{k})\right) \delta_p^T Q\delta_v - \tilde{V} \left(Q - k(k+\overline{k})Q - kI_{3\times 3} \right) \delta_p + (k+\overline{k})\delta_v^T Q\tilde{V} \\
&\quad -\beta\tilde{V}^T V_T
\end{aligned}$$

Once again apply Young's inequality to obtain conditions on gains to guarantee convergence.

$$\begin{aligned}
\dot{\mathcal{L}} &\leq -(k+\overline{k})\lambda_{min}(Q)||\delta_p||^2 - \left(k_2 - \overline{k}\lambda_{max}(Q)\right)||\delta_v||^2 - (\beta - k\lambda_{max}(Q))\,||\tilde{V}||^2 \\
&\quad +(1+\overline{k}(k+\overline{k}))\left(\epsilon_1||\delta_p||^2 + \frac{1}{4\epsilon_1}\lambda_{max}^2(Q)||\delta_v||^2 \right) + (\overline{k}+k)\left(\epsilon_2||\delta_v||^2 \right. \\
&\quad \left. +\frac{1}{4\epsilon_2}\lambda_{max}^2(Q)||\tilde{V}||^2 \right) + \left(\epsilon_3||\tilde{V}||^2 + \frac{1}{4\epsilon_3}||\Delta||^2||\delta_p||^2 \right) + \beta\left(\epsilon_4||\tilde{V}||^2 + \frac{1}{4\epsilon_4}||V_T||^2 \right) \\
&\leq -\left((k+\overline{k})\lambda_{min}(Q) - (1+\overline{k}(k+\overline{k}))\epsilon_1 - \frac{1}{4\epsilon_3}||\Delta||^2 \right)||\delta_p||^2 \\
&\quad -\left(k_2 - \overline{k}\lambda_{max}(Q) - (\overline{k}+k)\epsilon_2 - \frac{1+\overline{k}(k+\overline{k})}{4\epsilon_1}\lambda_{max}^2(Q) \right)||\delta_v||^2 \\
&\quad -\left(\beta - k\lambda_{max}(Q) - \frac{\overline{k}+k}{4\epsilon_2}\lambda_{max}^2(Q) - \epsilon_3 - \beta\epsilon_4 \right)||\tilde{V}||^2 + \frac{\beta}{4\epsilon_4}||V_T||^2 \qquad (4.21)
\end{aligned}$$

where, $\Delta = Q - k(k+\overline{k})Q - kI_{3\times 3}$. Assume that,

$$\begin{aligned}
(k+\overline{k})\lambda_{min}(Q) - (1+\overline{k}(k+\overline{k}))\epsilon_1 - \frac{1}{4\epsilon_3}||\Delta||^2 &= \alpha_1 \\
k_2 - \overline{k}\lambda_{max}(Q) - (\overline{k}+k)\epsilon_2 - \frac{1+\overline{k}(k+\overline{k})}{4\epsilon_1}\lambda_{max}^2(Q) &= \alpha_2 \\
\beta - k\lambda_{max}(Q) - \frac{\overline{k}+k}{4\epsilon_2}\lambda_{max}^2(Q) - \epsilon_3 - \beta\epsilon_4 &= \alpha_3
\end{aligned}$$

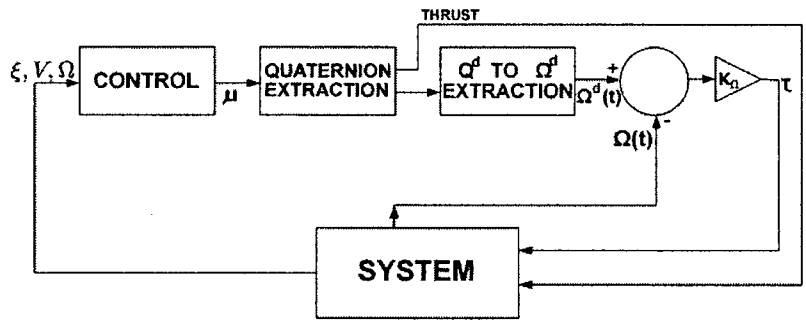

Figure 4.1: Block diagram of the control system.

Using the above simplifications, the time derivative of Lyapunov candidate function can be written as

$$\dot{\mathcal{L}} = -\alpha_1||\delta_p||^2 - \alpha_2||\delta_v||^2 - \alpha_3||\tilde{V}||^2 + \frac{\beta}{4\epsilon_4}||V_T||^2 \tag{4.22}$$

Therefore, for some positive Young's equation constants ϵ_1, ϵ_2, ϵ_3, and ϵ_4, the following three conditions must be satisfied to guarantee the boundedness of the errors δ_p, δ_v, and $\tilde{V}$ and their convergence to a domain $D = \left(\delta_p, \delta_v, \tilde{V}/\alpha_1||\delta_p||^2 + \alpha_2||\delta_v||^2 + \alpha_3||\tilde{V}||^2 \leq \frac{\beta}{4\epsilon_4}||V_T||^2\right)$

- $\alpha_1 > 0$
- $\alpha_2 > 0$
- $\alpha_3 > 0$

4.3 QUATERNION EXTRACTION

In the previous section, the force, μ, required to move the airborne vehicle closer to the target was derived. In order to apply the desired forces on the system, the attitude of the aircraft must be changed through changing the thrust and the torque. The quantity μ contains information about the desired thrust as well as the attitude and extracting this desired information properly is achieved through quaternion extraction. A secondary negative feedback controller is then

required to compare and minimize the difference between the desired angular velocity and the instantaneous angular velocity. A simple high-gain negative feedback loop is introduced in the control scheme inside the main control loop for this purpose as shown in Figure 4.1.

A quaternion is a four component vector of unit magnitude. The four components of the vector contain one scalar,(q_0), and one row vector, ($[q_1, q_2, q_3]^T$), i.e. $Q = [q_0, q_1, q_2, q_3]^T$. The force μ is given by

$$\mu \equiv gR\hat{z} - \frac{T}{m}\hat{z} \tag{4.23}$$

Substituting the relationship between rotation matrix R and quaternion Q given by

$$R = I_{3\times 3} + 2S(q)^2 - 2q_0 S(q)$$

into Eq. 4.23, it can be shown that the relationship between μ and quaternion Q is

$$\mu = g \begin{pmatrix} 2q_1q_3 - 2q_0q_2 \\ 2q_2q_3 + 2q_0q_1 \\ -\frac{T}{mg} + 1 - 2(q_1^2 + q_2^2) \end{pmatrix} \tag{4.24}$$

For the purpose of simplification, let $q_3 = 0$. The simplified relationship between μ and Q is then given by,

$$\mu = \begin{pmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \end{pmatrix} = g \begin{pmatrix} -2q_0q_2 \\ 2q_0q_1 \\ -\frac{T}{mg} + 1 - 2(q_1^2 + q_2^2) \end{pmatrix} \tag{4.25}$$

Let the thrust be as

$$T := mg(\alpha q_0^2 + 1 - 2(q_1^2 + q_2^2)) \tag{4.26}$$

with α being a variable the value of which will be determined later. This leads to

$$\begin{pmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \end{pmatrix} = g \begin{pmatrix} -2q_0q_2 \\ 2q_0q_1 \\ -\alpha q_0^2 \end{pmatrix} \tag{4.27}$$

There are three equations and three unknowns and the goal is to determine the values of q_0, q_1, and q_2. This can be done by either solving the last equation for q_0, or by using the first two equations. The first approach would result in singularity at $\mu_3 = 0$. Therefore, q_0 must be solved through the first two equations. Let,

$$\eta_1 = \mu_1^2 + \mu_2^2 = 4g^2 q_0^2 (q_1^2 + q_2^2) = 4g^2 q_0^2 (1 - q_0^2) \tag{4.28}$$

The solution of above equation can be found using the quadratic formula and is given by

$$q_0^2 = \frac{g + \sqrt{g^2 - \eta_1}}{2g} = \frac{\eta_2}{2g} \tag{4.29}$$

with $n_1 \leq g^2$. It can be easily verified that the range of q_0^2 is between 0 and 1. The other solution of q_0^2 is ignored because it will yield $q_0 = 0$ for $\eta_1 = 0$. If the conditions $|\mu_1| < 2g$ and $|\mu_2| < 2g$ are imposed, then the solution of q_1 and q_2 are

$$\begin{aligned} q_1 &= \frac{\mu_2}{2gq_0} \\ q_2 &= \frac{\mu_1}{2gq_0} \end{aligned}$$

Now that all the components of quaternion have been found, the value of α can be determined to guarantee that the thrust 'T' is always positive. By choosing

$$\alpha = -\frac{\mu_3}{gq_0^2}$$

the condition on μ_3 can be derived to keep the thrust positive all the time. Substituting the expressions of q_0, q_1, q_2, and α in Eq. 4.26, one gets

$$-\frac{\mu_3}{g}+1-2\left(\frac{2g\mu_2^2}{4g^2\eta_2}+\frac{2g\mu_1^2}{4g^2\eta_2}\right)>0$$

$$-\frac{\mu_3}{g}+1-\frac{\eta_1}{g\eta_2}>0$$

$$\frac{\mu_3}{g}<1-\frac{\eta_1}{g\eta_2}$$

$$\mu_3<g-\frac{\eta_1}{\eta_2}$$

Therefore, by calculating the components of quaternion using the formulae,

$$q_0 = \sqrt{\frac{\eta_2}{2g}} \tag{4.30}$$

$$q_1 = \frac{\mu_2}{\sqrt{2g\eta_2}} \tag{4.31}$$

$$q_2 = -\frac{\mu_1}{\sqrt{2g\eta_2}} \tag{4.32}$$

$$q_3 = 0 \tag{4.33}$$

where,

$$\eta_1 = \mu_1^2+\mu_2^2$$
$$\eta_2 = g+\sqrt{g^2-\eta_1}$$

and satisfying the following two conditions,

$$\mu_1^2+\mu_2^2 \leq g^2 \tag{4.34}$$

$$\mu_3 < g-\frac{\mu_1^2+\mu_2^2}{g+\sqrt{g^2-\mu_1^2-\mu_2^2}} \tag{4.35}$$

it can be guaranteed that a feasible solution exists. By satisfying the above conditions, the desired thrust can be extracted along with the four elements of quaternion which can then be used for obtaining the desired angular velocity Ω^d as shown in [30]. The control input τ_a is generated using a high gain feedback as follows:

$$\tau_a = K_\Omega\left(\Omega^d-\Omega\right) \tag{4.36}$$

The thrust extracted above along with the control torque τ_a are then fed back into the system's dynamical model to change the position of the aircraft.

Chapter 5

SIMULATION RESULTS

The simulation results for both controllers, 'Controller 1' and 'Controller 2' are presented in this section. The inertia matrix, I_b, used is $\text{diag}[0.5, 0.5, 0.25] N.m^2$, the initial conditions of the rotation matrix are $I_{3\times3}$. The mass of the system is 4.313kg while the vertical lever arm l is 0.1778m [17].

5.1 CONTROLLER 1

In the simulations of the first control law, it is assumed that the target velocity is known. A disturbance force of $F_d = -[1, 1, 0]^T N$ is applied at a distance of 0.1m above the center of the lens of the camera. The initial position of the camera is $\xi = [-1.3, 2, -4]^T m$ in $\mathcal{B}$ and the desired image features vector is $\sum p_i^* = [0, 0, 1.993]^T$ corresponding to a location of $[0, 0.75, -3]^T m$. There are two targets points, separated by a distance of 0.5m, moving with a constant velocity of $\pm[0.025, 0, 0]^T$m/s in a neighborhood of 2.5m on the floor. The first target point starts its journey at $[0, 0.5, 0]^T m$, travels linearly a distance of 2.5m in the x-direction, and then travels backward to its origin in the same fashion. The second target point starts its journey at $[0, 1, 0]^T$ and moves in the same way as the first target point in the x-direction. The values of the gains and other constants used are as follows:

- $k_1 = 15$
- $k_2 = 0.4$
- $\Gamma = 0.1$
- $k_\Omega = 200$

The control effort of Eq. 4.6 is applied to the aircraft and the simulations were ran for 480 seconds. The results are shown below.

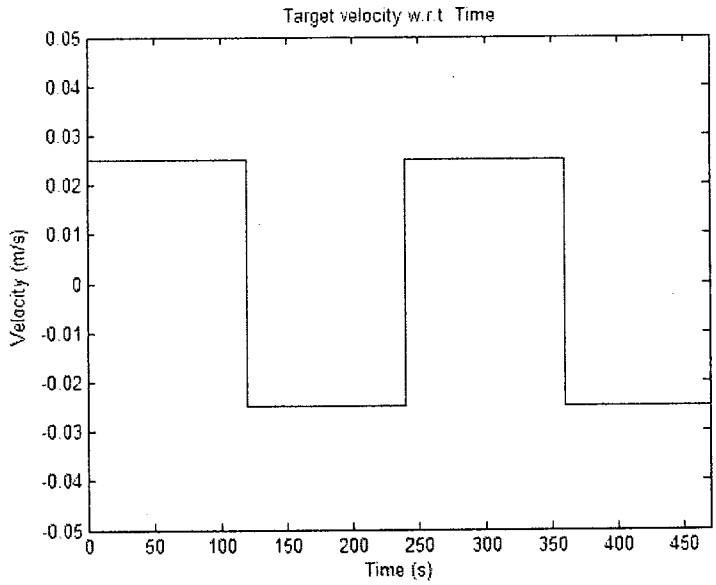

Figure 5.1: The velocity of target w.r.t. time.

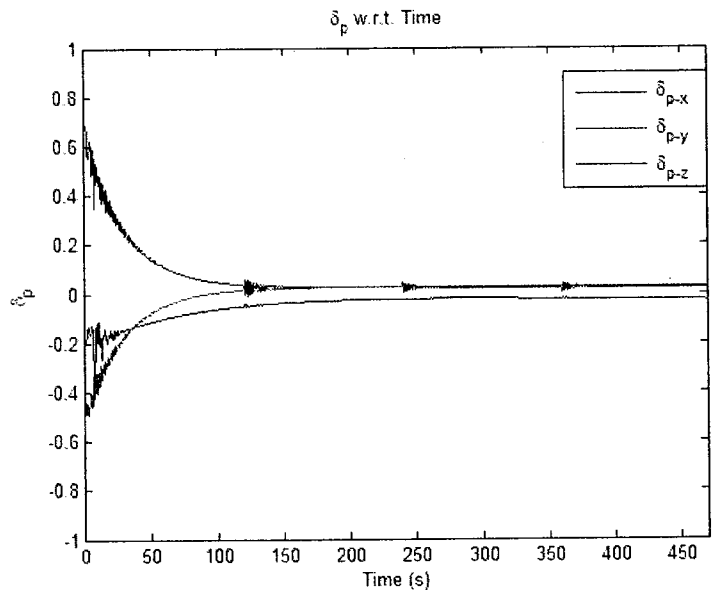

Figure 5.2: The three components of δ_p with known velocity of the target.

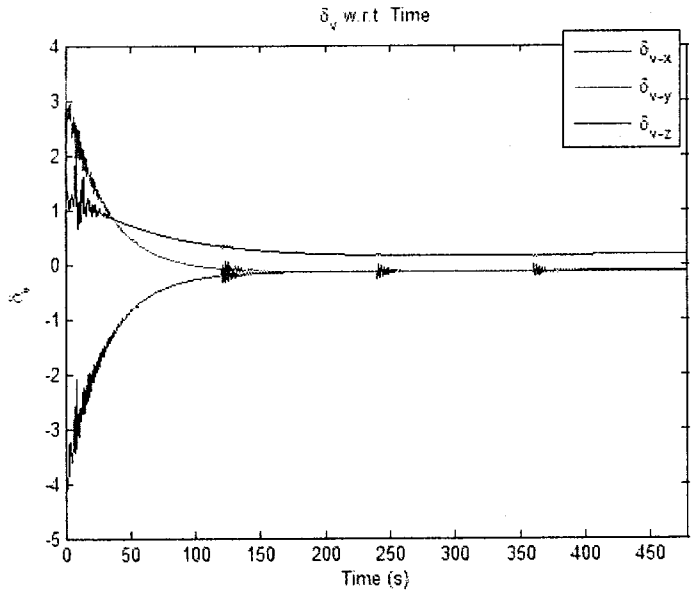

Figure 5.3: The three components of δ_v with known velocity of the target.

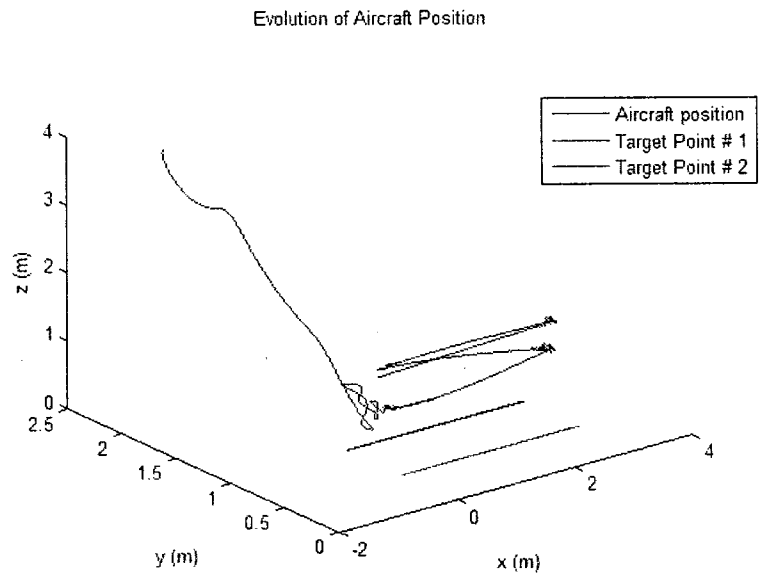

Figure 5.4: The position of the camera w.r.t. time.

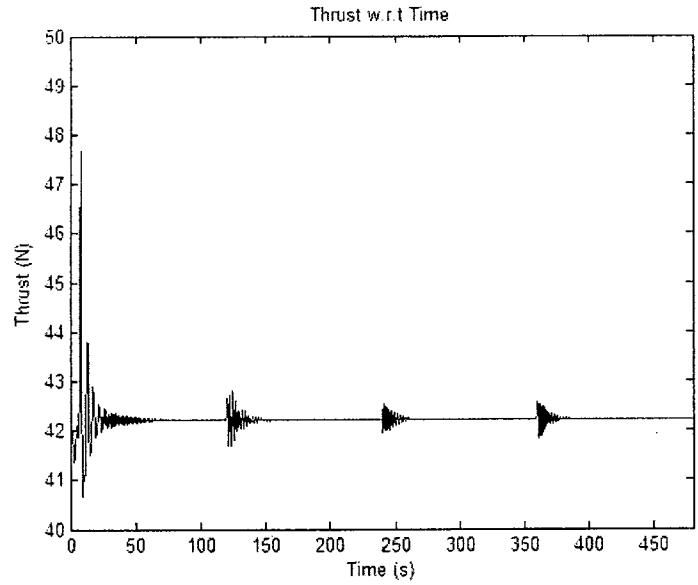

Figure 5.5: The magnitude of thrust w.r.t. time.

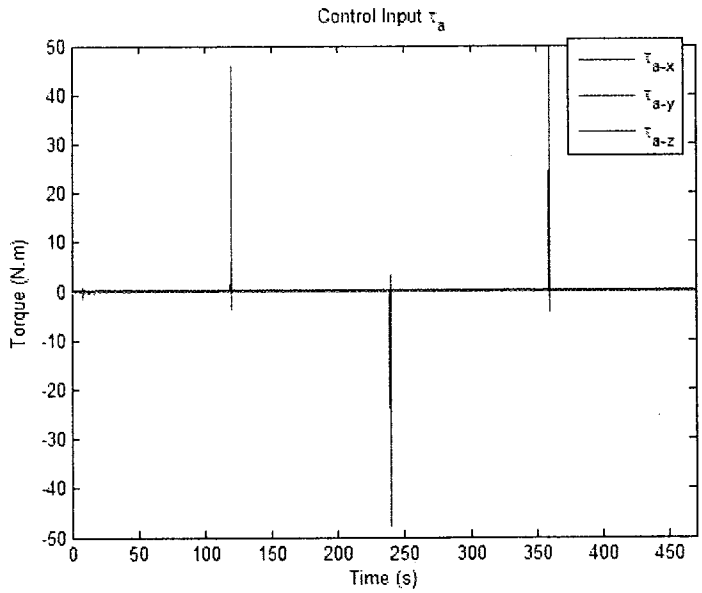

Figure 5.6: The magnitude of control torque w.r.t time

It can be noticed that the error δ_p does not converge exactly to zero which is due to the trade-off between the control torque and the error δ_p. In order to obtain convergence to zero of the error δ_p, much greater values of control torque, τ_a, are required especially when the target switches its velocity. The results shown above are a good balance of the required control torque and the error δ_p.

The 'x' and 'y' components of the error δ_p converge faster than the 'z' component. It takes 320 seconds for x,y, and z components of δ_p to settle to their final values, i.e. $[0.025, 0.025, -0.023]^T$. Same is the case for δ_v where x and y components converge to their final values faster than the z component. The steady state error for δ_v is $[0.13, 0.15, -0.15]^T$. The thrust takes 90 seconds to converge to 42.2N, which is also its steady state value.

It can be seen in the above simulation results that there are repeated oscillations at 120 seconds interval in δ_p, δ_v, and thrust along with spikes in the control input τ_a. This occurs when the target changes its direction instantaneously and the UAV has to move in opposite direction all of a sudden which requires high torque momentarily. Once the UAV has changed its direction, it takes approximately 25 seconds for the thrust and the two errors (δ_p and δ_v) to settle down to their final values.

5.2 CONTROLLER 2

The control effort expressed in Eq. 4.19 is applied to the aircraft. The initial position of the camera is $-[3,3,9]m$ in $\mathcal{B}$, while the desired image features vector $\sum p_d^* = [0,0,1.985]^T$ corresponding to a location of $[0,-1,-8]^T m$. The initial position of the first target point is $[0,0,0]^T m$ while the initial position of the second target point is $[0,-2,0]^T$. Both target points are moving linearly at a constant velocity of $-[0.05,0.05,0]^T m/s$ in $\mathcal{B}$. A disturbance force, $F_d = -[1,1,0]N$ was applied at a distance of 0.1m below the center of lens of the camera. The values of the other gains are as follows:

- $\overline{k} = 1.5$
- $\beta = 0.75$
- $k = 0.01$
- $k_2 = 50$
- $\Gamma = 0.01$
- K_Ω= 20000

The simulations were ran for a time period of 600 seconds. The transition responses are also shown. The results are as follows.

It can be seen from the simulation results that the error δ_p has a very small steady-state error which happens due to the fact that magnitudes of both the target velocity V_T and the gain β are quite small. The error δ_p takes about 165 seconds to converge to its final value, i.e. $[-0.011,-0.016,-0.002]^T$. The error δ_v takes about 25 seconds to converge to its final value, i.e. $[-0.033,-0.025,0.004]^T$. The error $\tilde{V}$ takes about 20 seconds to converge to $[0.05,0.05,0]^T$. The thrust settles to it's final value, i.e. 42.2N, in less than one second.

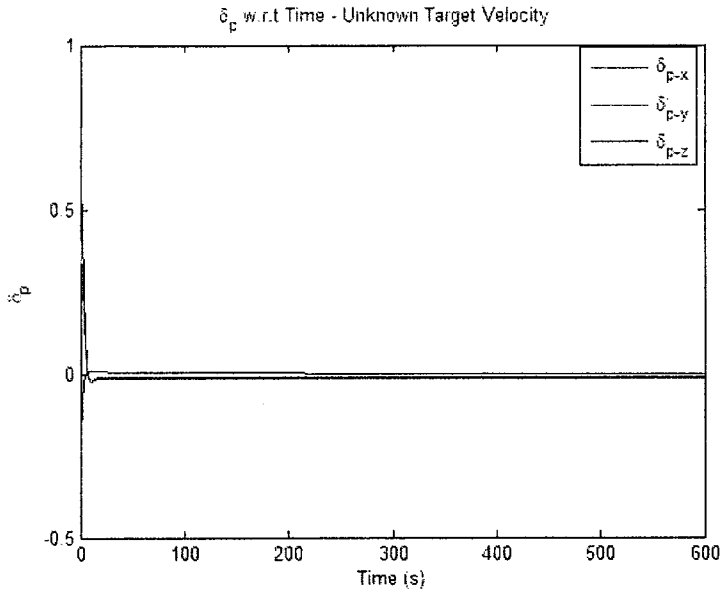

Figure 5.7: The x,y,z components of δ_p with unknown V_T.

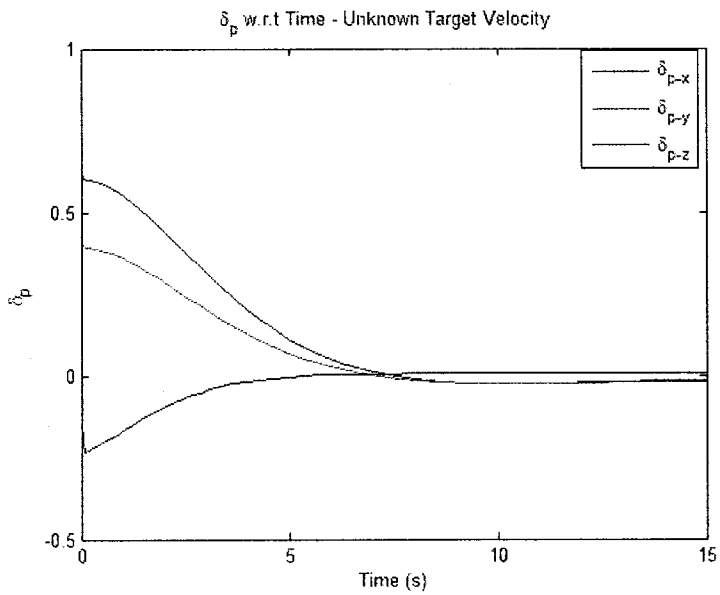

Figure 5.8: The transition of x,y,z components of δ_p with unknown V_T.

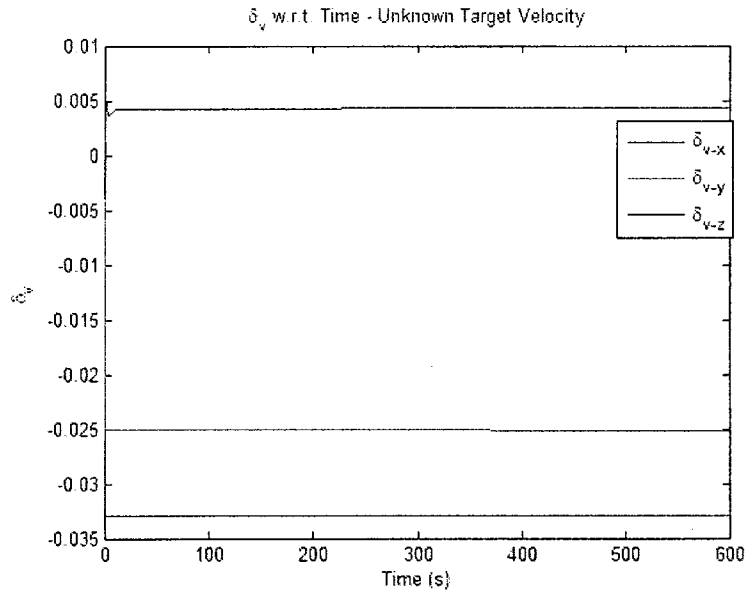

Figure 5.9: The x,y,z components of δ_v with unknown V_T.

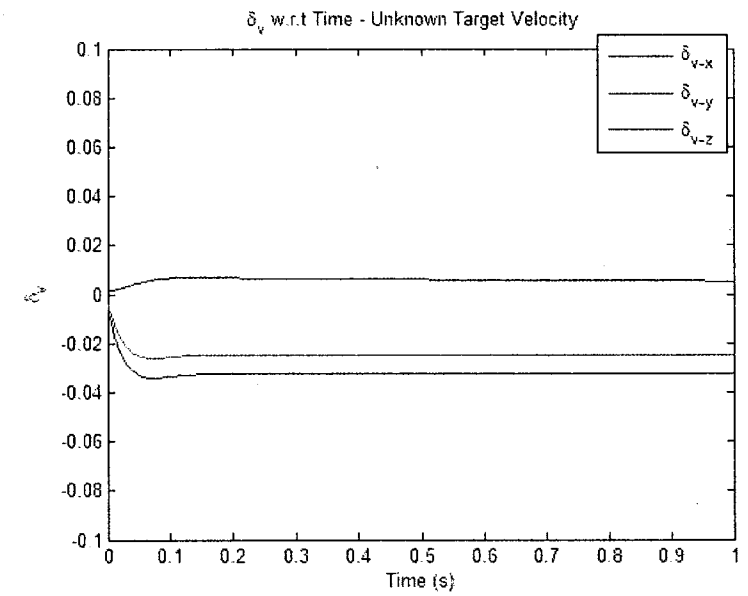

Figure 5.10: The transition of x,y,z components of δ_v with unknown V_T.

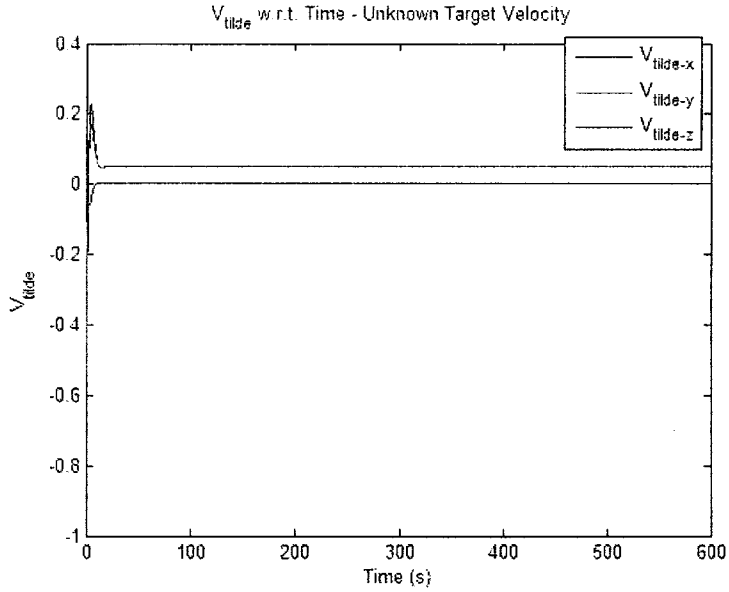

Figure 5.11: The x,y,z components of $\tilde{V}$ with unknown V_T.

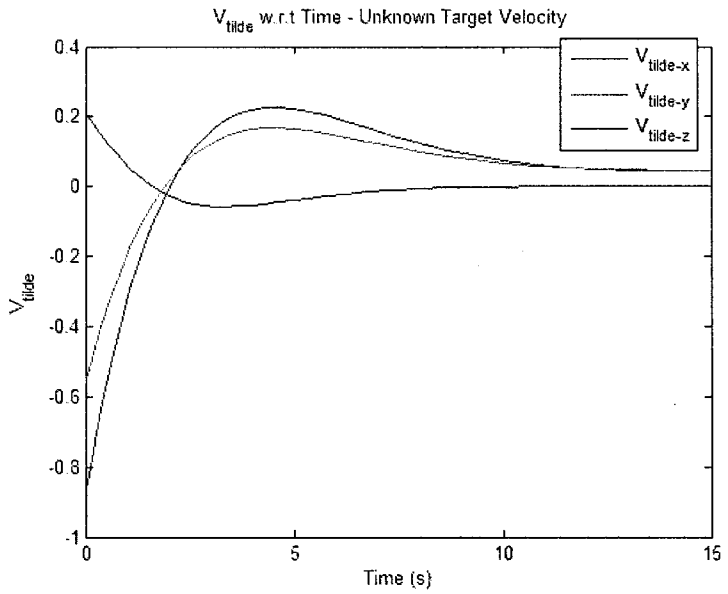

Figure 5.12: The transition of x,y,z components of $\tilde{V}$ with unknown V_T.

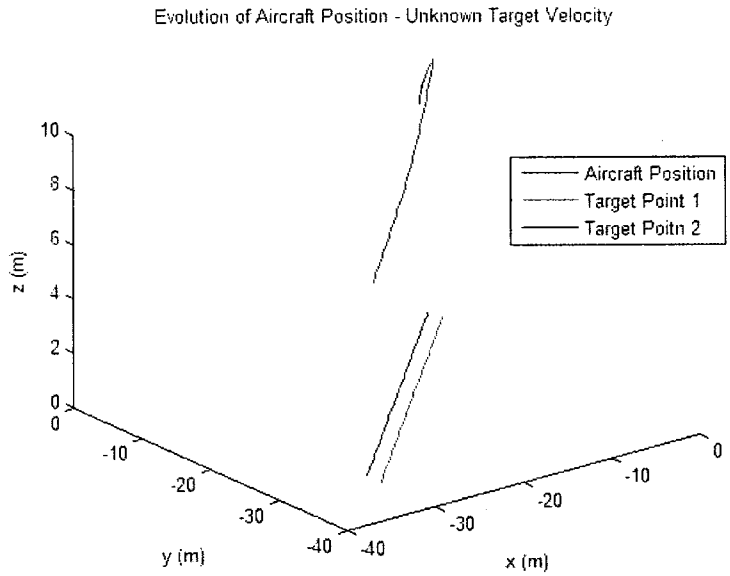

Figure 5.13: The evolution of camera position w.r.t time - unknown V_T.

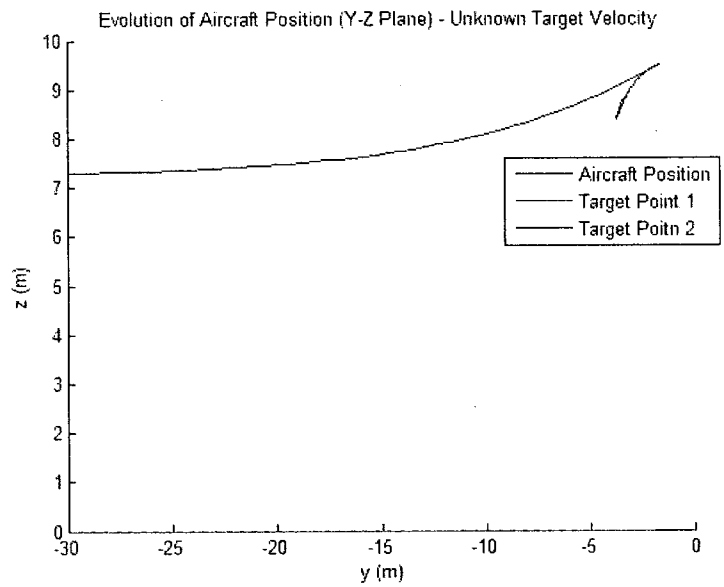

Figure 5.14: The evolution of camera position on Y-Z plane w.r.t time - unknown V_T.

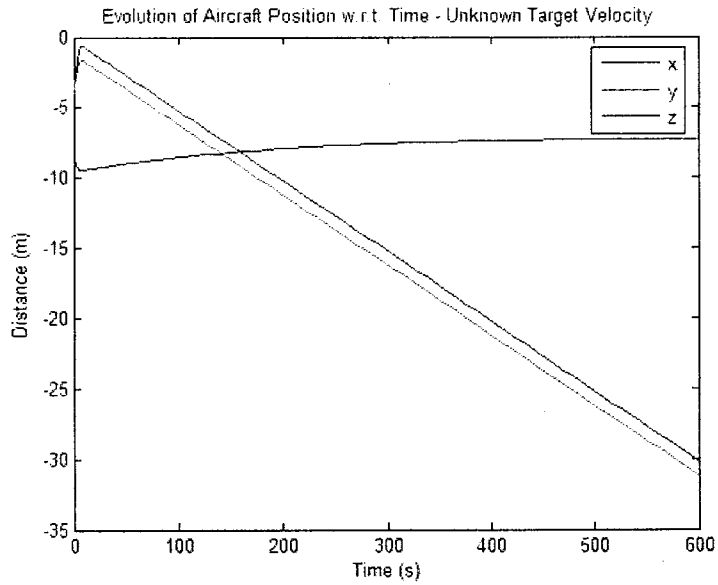

Figure 5.15: The evolution of camera position on 2D plane - unknown V_T.

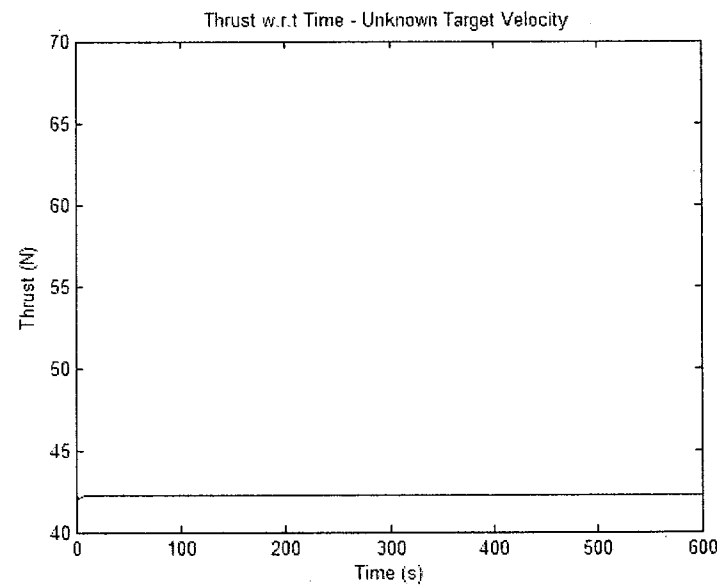

Figure 5.16: The magnitude of thrust w.r.t. time - unknown V_T.

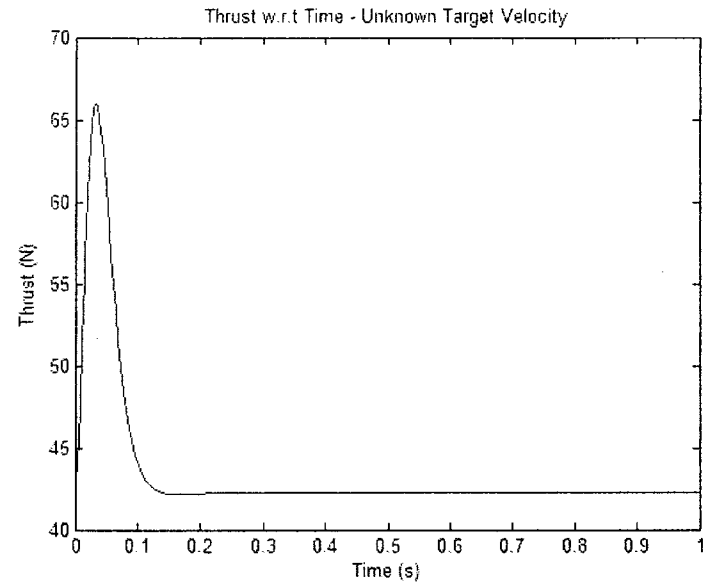

Figure 5.17: The transition of thrust w.r.t. time - unknown V_T.

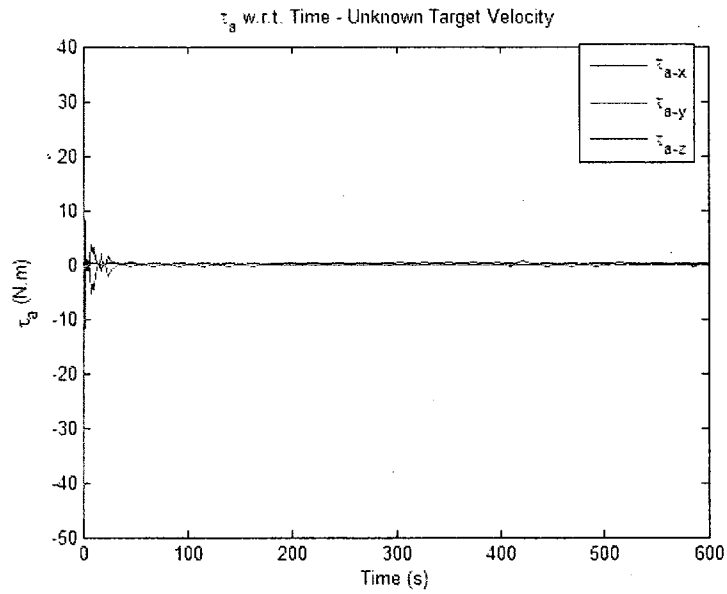

Figure 5.18: Control torque w.r.t. time - unknown V_T.

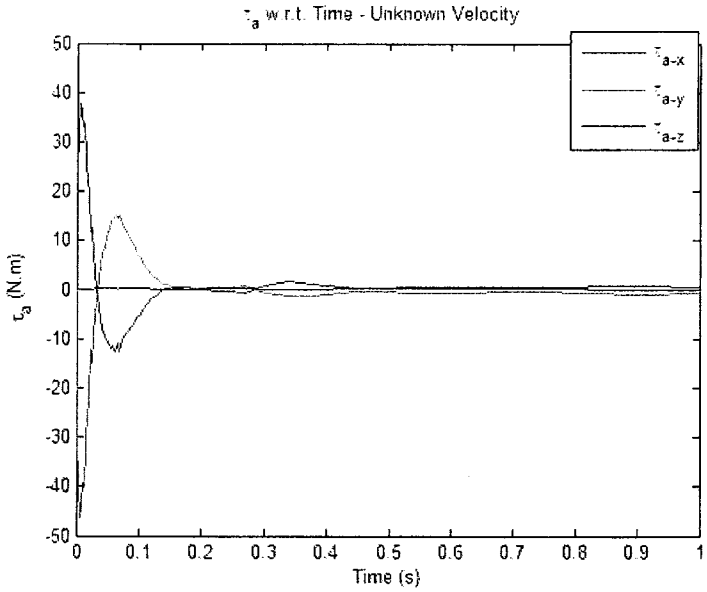

Figure 5.19: The transition of control torque w.r.t. time - unknown V_T.

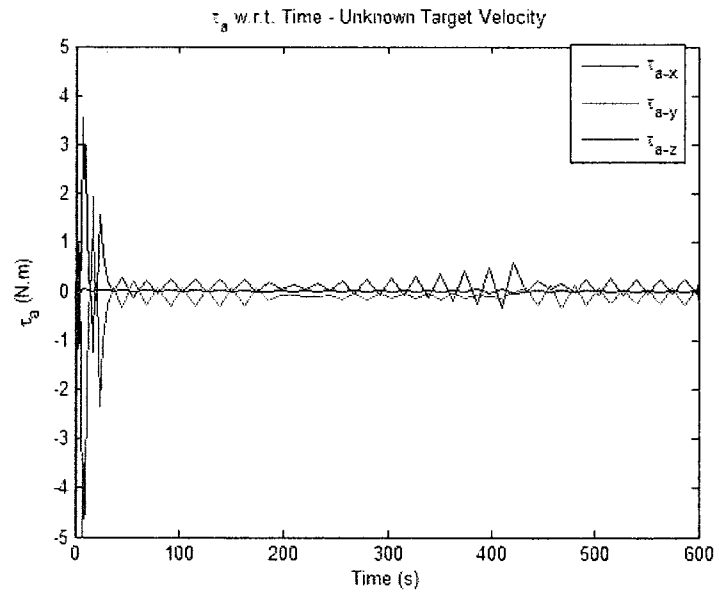

Figure 5.20: Close-up of control torque w.r.t. time - unknown V_T.

Chapter 6

EXPERIMENTAL SETUP

A scale-model prototype of a ducted fan VTOL UAV was designed and developed for experimental evaluation of the two controllers proposed. The UAV design objectives had two aspects: one mechanical and one electrical. The objective of the mechanical design was to design a system with maximum possible strength without sacrificing the weight. This was achieved by using composite materials like carbon fibre or fibre-glass and other lightweight materials like aluminum or plastic. The electrical design objective was to develop a fully autonomous system capable of controlling all aspects of flight dynamics along with calculating control effort on-board to achieve the desired task. To achieve this objective, modern state-of-the-art sensors, servos, camera-board, sensor node and other electronics were used.

6.1 MECHANICAL SYSTEM DESIGN

Broadly speaking, the system can be segregated into three distinct portions, namely the nose cone, the shroud or the duct, and the tail cone. The duct can be thought of as the fuselage of the UAV, whereas the nose cone and the tail cone are required to house different sensors, electronic boards, and the system power plant. The CAD drawings pertaining to different individual parts of the system are provided in Appendix A, and the overall system description is given below.

The nose cone is the top most structure, resembling a shelf made of aluminum bars. The best way to attach different levels of shelves is through small screws because welding the thin aluminum bars may deform them. The lowest level of the shelf is planned for storing one of the two batteries that powers the propellers. On the next level reside an Electronic Speed Controller (**ESC**) and another battery used to power sensitive electronic equipment. On the next level, one can find the Imote2 board, and the level above it would contain Iner-

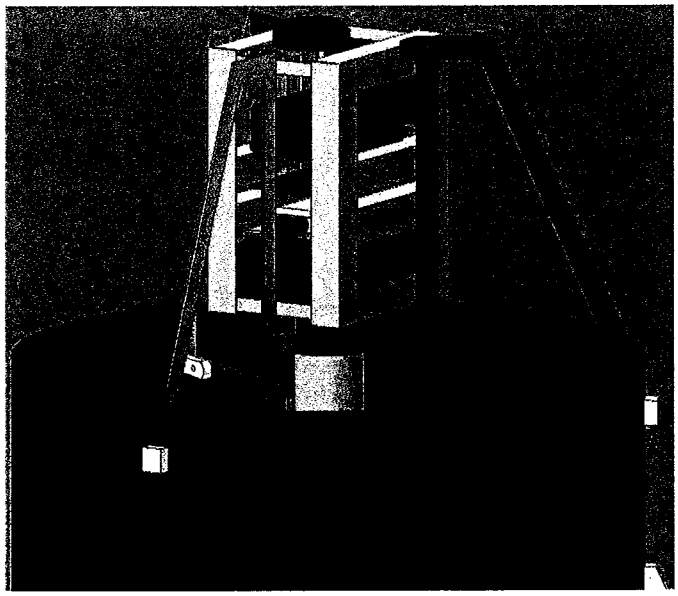

Figure 6.1: The nose-cone of the system attached to the duct.

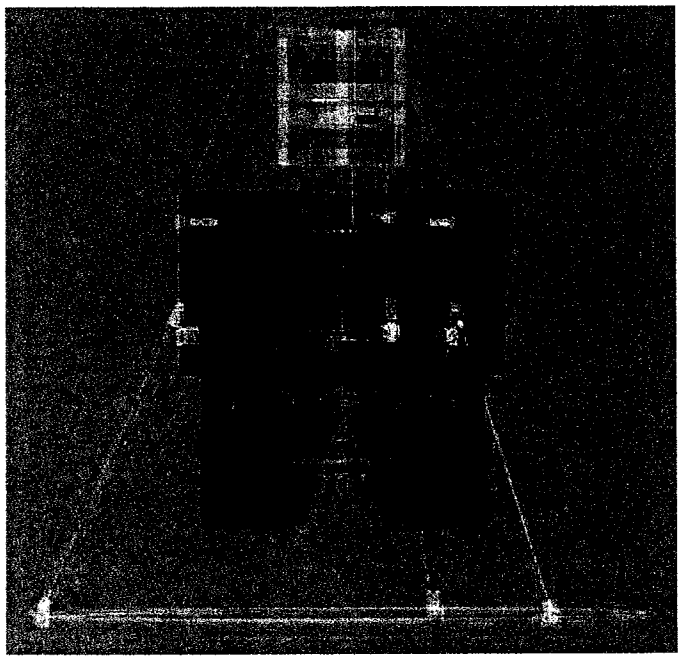

Figure 6.2: The cross-section of UAV with the body

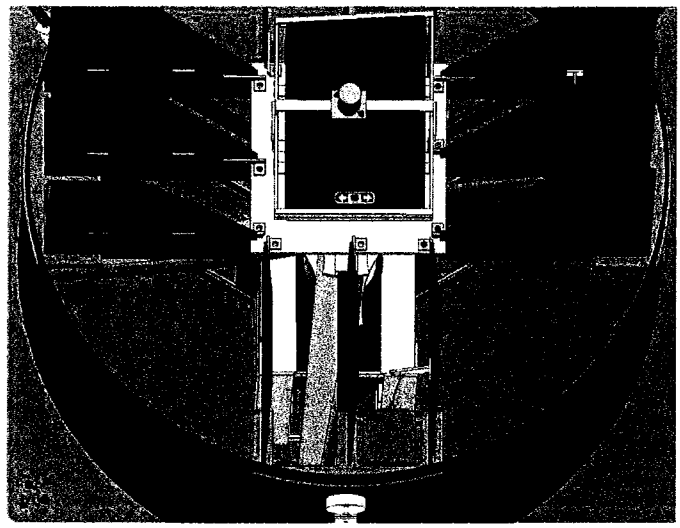

Figure 6.3: The tail-cone and the ailerons of the UAV

tial Measurement Unit (IMU). The GPS antenna is attached on top of the nose cone. The nose cone is attached to the main fuselage using three triangles made of fibreglass. Three different kinds of custom designed brackets are attached to each of the three sides of the triangle. One end of the triangle is attached to the nose cone, the other one is attached to the duct, and the third end is attached to the motor mount plate.

The duct, also called shroud, is made using carbon fibre. Four layers of 3K-Plain-Weave carbon fibre fabric from FibreGlast Corporation were used to manufacture the duct. Once manufactured, the holes of different sizes were drilled on a milling machine 120 degrees apart in addition to the four 0.25 inch slots that were drilled 90 degrees apart. The two propellers are enclosed inside the duct. The propellers are attached to two separate motors, each of which is held using an aluminum plate. Due to the fact that drilling holes in the carbon fibre tube greatly reduces its strength, the lower motor mount is different in shape than the top one. The lower motor mount has a circular tube like opening to facilitate the insertion of carbon fibre tubes. The top motor mount is held in place using brackets that attach it to the nose-cone-holding triangle. The lower mount is held using carbon fibre tubes that are inserted in a two-piece plastic bracket attached to the duct.

At the exit of the duct, one can find the ailerons, also called flaps Fig.6.3. Each aileron is made out of 0.0625in thick fibreglass. There are twelve ailerons in total with a set of three ailerons on each side. The ailerons are controlled through servo that is mounted on the same bracket as the ailerons. Each set of

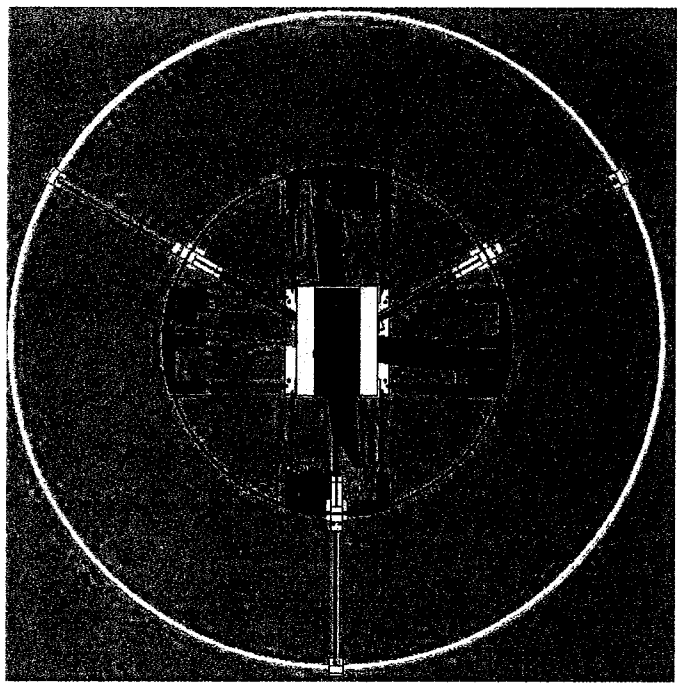

Figure 6.4: The view of UAV from landing ring.

ailerons is actuated by a HiTec HS-81 micro servo, and due to this fact, all three ailerons of each set are connected together through plastic linkages. The servo is connected to the aileron using a plastic ball-joint along with a metal linkage. The length of metal linkage is designed to allow flaps to deviate a maximum of 30 degrees at either side.

The ailerons, as shown in Fig.6.3, are the primary control surfaces used to change the attitude of the aircraft as they are used to control the direction of the outgoing airflow. Each aileron has two parts: one fixed and the other one that tilts. These two parts are connected together using hinges. One end of the fixed part of the aileron is used to attach the aluminum plate that holds the tail cone. The tail cone is a two piece aluminum structure that holds the battery, sonar, a circuit board along with another ESC to control the speed of propellers. The UAV is designed to sit on a circular landing ring, shown in Fig.6.4, made of poly-vinyl-chloride (PVC) tube. There are two specially designed brackets called Landing-Ring-Shoes used to attach this ring to the duct via a carbon-fibre tube link. The carbon fibre tube is inserted in a bracket attached to the duct through screws at the other end. There are four holes for each bracket to distribute the impact of the force in case of an accident. There are three legs in total, separated at an angle of 120 degrees. These brackets would require cus-

tom machining and have not been manufactured right now due time limitations.

6.2 ACTUATORS

There are two motors, each of which is attached to two counter rotating rotors. Due to the fact that this UAV building project is the continuation of the work documented in [17], a lot of components have been carried over from that project. The actuators used to rotate propellers are the same as documented in [17], i.e. Plettenburg Orbit 25-12 outrunner motor. This motor is a three phase motor with a maximum speed of 25,000 rpm. Two, three-blade propellers were used that were slightly different, with the main difference between the two propellers being the pitch angle. The upper propeller is called the pusher type propeller and the lower propeller is called the tractor type propeller. Both propellers have the same diameter of 13.869 in.

6.3 POWER

The system power plant was adopted from the work done in [17]. Three rechargeable batteries were used to power all the electronics and the motors. The battery used to power the motor and the servos is Thunder Power 6-cell 4.2Ah lithium-polymer battery capable of supplying continuous current of 63A with a peak current of 100A. In order to obtain sufficient operation time to conduct experiments, the power requirements demanded two separate batteries of this kind. The Thunder Power TP1010C charger and TP210V balancer were used to recharge batteries due to the sensitivities involved in using lithium-polymer batteries.

A separate low-voltage battery was required to power sensitive electronics due to the fact that most sensors and electronic boards operate around or 5V. The lithium-polymer batteries supply a voltage of at least 20V, and use of regulators would require bulky heat-sinks to dissipate the extra power. For the sake of saving weight, a separate 3-cell 800mAh lithium battery was used in addition to the main batteries above.

6.4 SENSORS AND OTHER AVIONICS

A variety of electronic boards and sensors can be found on-board the aircraft that are used not just to obtain instantaneous information about the aircraft but also to control its flight. Thanks to the numerous groups and researchers who are constantly working on improving avionics equipment, there are lots of ready made off-the-shelf components available that can be integrated into any project. Considerable time and efforts were put on investigating similar projects by other

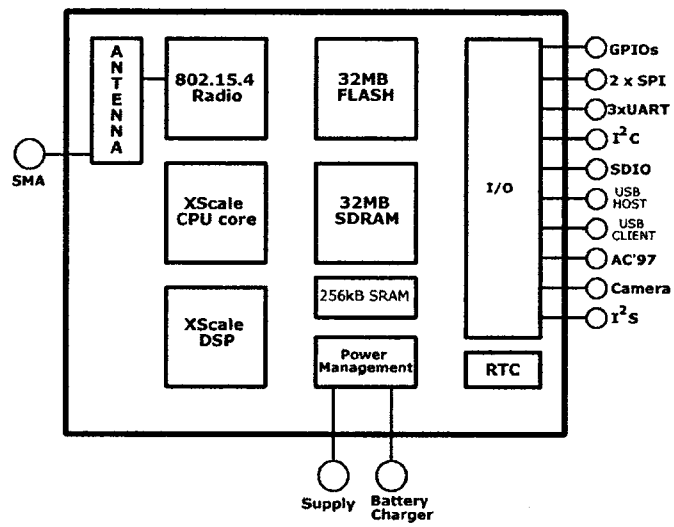

Figure 6.5: The Imote's IPR-2400 architecture.

research teams to obtain the best possible solution. The avionics comprises of three sensors, namely IMU, GPS, and sonar, a sensor node called Imote2, two Electronic Speed Controllers (ESC), and a custom designed electronic board to manage power and communication between various components. The details of these components are provided below.

6.4.1 The Imote2 Sensor Node

The sensor node, called the Imote2, is at the heart of all the avionics on-board the UAV. It is more than just a sensor node as it comes with various kinds of memory as well as a DSP coprocessor. The DSP coprocessor is used to process the algorithm designed for controlling aircraft motion and also to provide control signals to the system motors and servos. The particular model of Imote2 board selected for this project was Crossbow's IPR2400, along which the Imote2 interface board, IIB2400, and Imote2 multimedia camera board, IMB400CA, were also chosen. The IPR2400 board comes with eight different I/O ports as shown in Fig.6.5, utilizing five different communication protocols, namely UART, SPI, I2C, SDIO, and GPIO, along with dedicated I/O ports for camera chip interface, JTAG, etc.

Two of the three UART ports will be used for communication between IMU,

GPS, and the IPR2400 board. The remaining UART port and one SPI port would be used for communication between the microcontroller and the IPR2400. The remaining SPI port is reserved for communication between ESC interface and the IPR2400.

The Imote2 multimedia camera board, IMB400CA, can be connected to IPR2400 using the Camera Chip Interface. IMB400CA is just an expansion board used for capturing images and video. The processing of visual data can only be performed on IPR2400 board. Because the current version of IMB400CA board is supported under TINYOS, the programming for image processing would be done in nesC language.

The IIB2400 is an interface board that can be used to program and debug both the IPR2400 board as well as IMB400CA board. The interface board can be connected to a PC via a mini-B USB connector. In such a case, the IEEE 802.15.4 radio transceiver built-into the IPR2400 can be used for communication between IPR2400 board and IIB2400 board.

In the following sections, the term Imote2 refers simply to the IPR2400 board.

6.4.2 Inertial Measurement Unit

The inertial measurement unit (IMU) is a sensor assembly that gives the information about the attitude of the aircraft. IMUs are generally an assembly of three kinds of sensors, namely accelerometer, magnetometer, and gyroscopes. The particular IMU selected for this project was MicroStrain 3DM-GX1 and it contains three sets of each of the above mentioned sensors along with an A/D converter and an embedded microcontroller. Due to on-board filtering of the sensors' outputs, this IMU has a good built-in shield against magnetic field perturbations. This IMU can provide static as well as dynamic orientation and has the ability to output it either as Euler angles, quaternion, or orientation matrix.

The IMU requires a minimum DC voltage of 5.2V and draws 65mA current. It can transmit/receive the data using either RS-232 or RS-485 communication protocols and due to this fact it will be connected to the Imote2 board using UART communication protocol. The serial data communication rate of 19.2 k-baud should be used to synchronize communication with Imote2.

6.4.3 Global Positioning System

The Global Positioning System (GPS) gives information on the position of the UAV. The GPS sensor selected for this project is from NovAtel's SUPERSTAR II family, with SS-II 3-5Hz being the particular model. This GPS system utilizes Satellite Based Augmentation System (SBAS) for improved accuracy and measures position as well as velocity that are output between 3Hz and 5Hz

frequency. Since the UAV is designed for laboratory experimental purposes, the private output (PVT) was chosen over carrier phase measurement output. There are two versions of this GPS system, one that operates at an input voltage of 3.3V and the other one at 5.5V. The former one was chosen to save power and prolong battery life. The GPS would communicate with the Imote2 board using UART communication protocol.

The NovAtel SUPERSTAR II family of products are designed to operate using either active or passive antenna. The Wi-Sys-WS3997 surface mount GPS antenna was selected due to the fact that it can provide dependable, clear, and consistent GPS signal reception. It can operate between $2.7V_{DC}$ and $5V_{DC}$ and draws current between 9mA and 15mA. The GPS and the antenna are connected using antenna RF cable with BNC female connector at one end.

Since the controllers designed in this work require the measurement of position at the center of the camera, care must be taken to adjust the calculations for the difference in distance between the GPS antenna and the camera. The distance can be measured once the UAV is completely assembled.

6.4.4 The Sonar

The sonar is used to find out the height the UAV is at, or in other words the distance between the floor and the UAV. The sonar is integrated for future projects and is not required for the practical application of the control design documented in this work.

The sonar selected for the UAV is Deventech's SRF-02 which has a sensing range between 0.15m and 6m. It operates at $5V_{DC}$ (can tolerate up to $5.5V_{DC}$ inputs), and draws a current of 4mA. The sonar is designed to communicate using either I^2C bus or UART; the former one was chosen to save the third port of Imote2 for other applications. The sonar can be connected either directly to the Imote2 board or its data can be routed through a microcontroller to the Imote2 board if the intent is to save the Imote2 ports for expansion of other devices in the future.

6.4.5 The R/C Receiver

Even though the design objective was to build a UAV for autonomous operation with minimum pilot control, a radio-signal control option was added just in case for future purposes. The Futaba R168DF 8-channel FM receiver was added to receive input signals from pilot. The R/C receiver and transmitter are carried over from previous project documented in [17]. The R/C receiver outputs four pulse-position-modulated signals that can be routed to the Imote2 board through a microcontroller. The output port of the Imote2 board can be programmed to change the attitude of the aircraft using the input signals from the pilot.

6.4.6 Custom Designed Board

In order to manage communication and power of different sensors, servos, and sensor node, a custom designed electronic board would be required. The major devices on this board would be voltage regulators, microcontrollers, and opto-isolator. The voltage regulators are required to provide a stable output voltage of either $3.3V_{DC}$ or $5.2V_{DC}$ from a lithium-ion battery, the voltage of which varies between $12.3V_{DC}$ to $11.1V_{DC}$. A combination of switching and linear voltage regulators would be suitable since a switching regulator generates a lot of noise and linear ones require big heat sinks due to the high voltage drop. The number of voltage regulators required would depend upon the power requirements of different rails to which different components are connected.

The control of aircraft using radio-signals or the control of servos to change attitude require the use of microcontrollers. Two PIC24FJ64GA002 microcontrollers from Microchip Inc. were chosen for this purpose. The first microcontroller is used to take input signals from R/C receiver, the battery, and the sonar and transmits these signals to the Imote2 board through Serial Peripheral Interface (SPI). The second microcontroller is dedicated solely for controlling servos. It takes input from the Imote2 board and performs pulse width modulation on the input signals to control servos so that the desired force can be generated. The four PWM output signals are then fed into opto-isolators. The output of the opto-isolators is used to control the servos. The servos introduce excessive noise and transients that can effect sensitive electronic equipment and because of this, the two ground planes are separated using opto-isolator. Lastly, an external oscillator is also required for the two microcontrollers since using the internal oscillator of the microcontrollers may cause the signals to drift. A 7MHz crystal oscillator was used for this purpose.

Chapter 7

CONCLUSION

Aircraft control has been an area of research even before the invention of the first aircraft. After the first successful human flight by Wright brothers, the significant research activity has led to aircraft control being a sophisticated task comprising of various control subsystems. With advances in technology, the researchers are up to take aircraft control to new heights: to build aircraft capable of achieving the desired mission with minimum human input, if at all. New control algorithms and strategies are under investigation that can control not just flight dynamics but also accomplish the mission simultaneously. Visual servoing is one such strategy in which a camera is used as a sensor to control the motion of unmanned aircrafts.

Over the years, visual servoing has evolved into two main branches. The two-camera approach is called the Position-based visual servoing, while a single camera approach is called Image-based visual servoing. While both approaches have their advantages and drawbacks, the major disadvantage of IBVS approach is the loss of depth coordinate due to the fact that a 3D scene is projected on a 2D surface. This weakness of IBVS control design coerced the research team of Amidi, Kanade, and Miller to assume that the height of helicopter flight is constant at one meter throughout the experiment [24]. To overcome this weakness, several strategies have been proposed by different research teams with the latest techniques relying on a different coordinate system than the Cartesian coordinates. One such technique was developed by Hamel and Mahony in their work [25], in which they used spherical projection of the target, and was chosen for this work. The IBVS controllers designed thus far using spherical projection are designed for stationary targets. An attempt was made to take the IBVS control design using spherical projection to a new stage by designing controllers capable of chasing a moving target.

The dynamic model of the aircraft was taken from the existing literature. In the conventional dynamic models of aircraft, the translational and rotational dynamics are coupled through a common control input which makes the control

design more sophisticated. To overcome this problem, the change of coordinates technique was adopted from [28] and applied to remove the control input from the translational dynamics expression. The resulting dynamical model was used to design the control law later in the stages.

Due to the fact that the existing literature of image-based visual servo control of UAVs using spherical projection is on stationary targets, a new error model was required for moving targets. The dynamics of the position error, δ_p, are derived from first principles assuming that the desired image features move on the image surface due to the ego-motion of the camera only. This assumption helped in simplifying the analysis but it limits the practicality of the controllers and work must be done in the future to address this drawback.

Two control laws are derived, capable of controlling the motion of the UAV in 3D. Due to the fact that the depth of the target is unknown, the velocity that the target is moving at cannot be estimated using dynamics of image features. For this reason, the velocity of the target is either assumed to be known *a priori*, or is unknown. For the former case, a simple dynamic control law is derived to obtain the desired force necessary to obtain the exponential convergence of the position error to zero. For the latter case, an adaptive control law is derived to minimize the error. In both cases, the designed force was used to extract the desired thrust and the desired quaternion through quaternion extraction. The desired orientation was then used to obtain the three components of the desired angular velocity using the approach shown in [30]. A simple high gain proportional controller was used to generate the control torque using the difference between the desired and the instantaneous angular velocities.

Simulation results are presented to validate the control laws and to evaluate its performance. In both cases, the aircraft starts its flight at a distance away from the target, moves closer to the target, and then hovers along its path.

For the purpose of experimental evaluation of the controllers, a design of ducted-fan type VTOL UAV was developed using CAD tools. The avionics and the sensors were carefully selected and purchased for the project. The work on system building was initiated but could not be completed due to time restrictions. The main fuselage of the system, the duct, is ready but to hold the nose cone and the tail cone, custom made brackets are required. To carry on this project to the next stages, future work should commence at manufacturing these parts. Once the manufacturing is done, the mechanical system assembly should be performed, followed by electrical assembly. The final stage would comprise of programming the controller as well as the image processing algorithms on-board to evaluate the controller performance.

Appendix A

UAV PARTS

A.1 The Duct

The duct, as shown in Fig.A.1 is made using composite material called carbon fiber. The duct was made using 4 layers of 3K, Plain Weave Carbon Fiber fabric. The duct is 8 *in* long and has a diameter of 14.1 *in*. It has square and circular holes drilled into it 120^o apart to allow attachment of different brackets. In addition to that, it also contains four slots separated at 90^o angle to attach ailerons.

A.2 The Nose Cone

The nose cone is made using light weight aluminum bars and is attached to the duct using triangles shown in Fig.A.3. These triangles are made using fiber glass. These triangles are connected to the duct using brackets shown in Fig.A.4. The brackets are made out of light weight plastic.

A.3 Inside The Duct

A pair of propellers rotating in opposite directions is the main component inside the duct. The two motors that propellers attach to are held by a pair of aluminum plates. Both of these plates are different in shape due to some limitations in available components. The top motor mount is shown in Fig.A.5 while the bottom motor mount is shown in Fig.A.6. The thickness of each motor mount is 0.125 *in*. The top motor mounts are attached at the end of triangles using ordinary vertical brackets while the lower motor mounts have a special structure to facilitate the insertion of carbon fiber rods. A specially designed two piece plastic bracket set shown in Fig.A.7 and Fig.A.8 is attached in the duct and

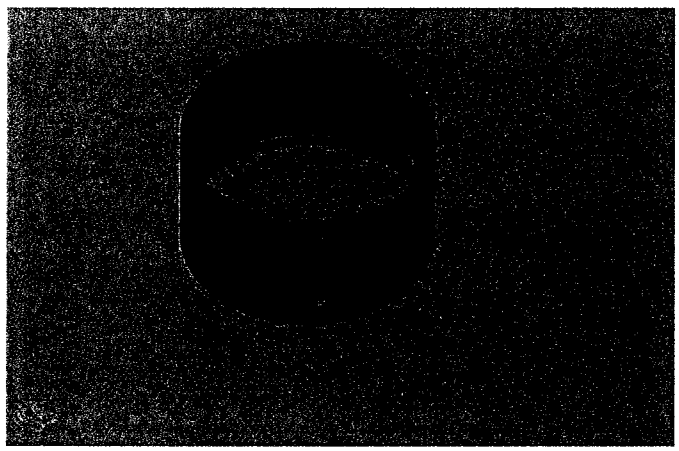

Figure A.1: The duct.

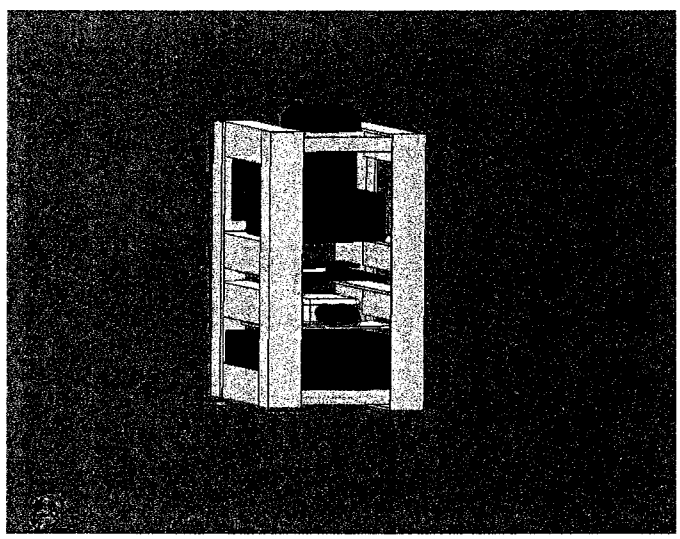

Figure A.2: The nose cone

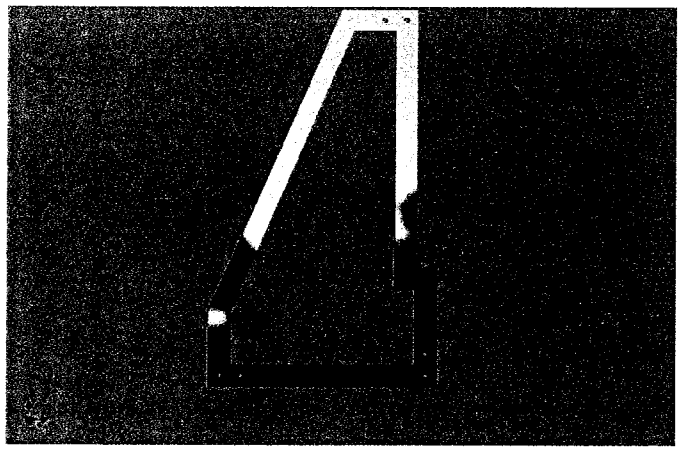

Figure A.3: The nose cone connecting triangle

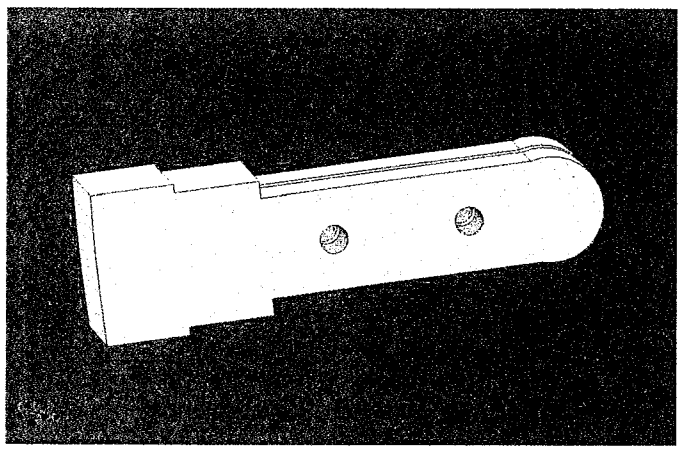

Figure A.4: Bracket to attach nose cone triangles

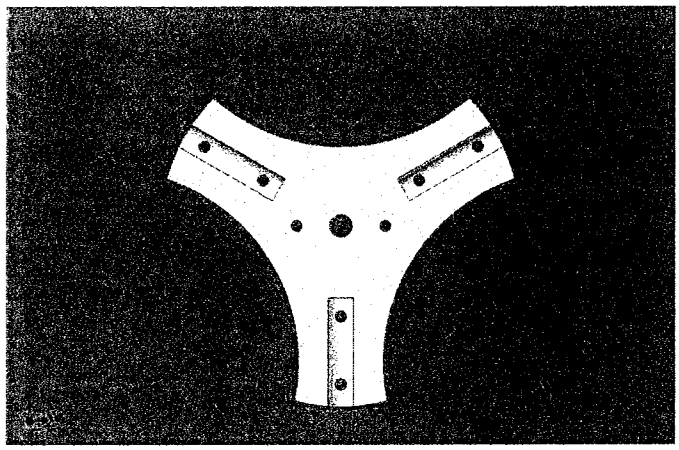

Figure A.5: The top motor mounting plate

carbon fiber rods are used to link the brackets to the lower motor mount.
The two pieces of the plastic bracket are attached to each other using screws. Both of the brackets are curved at one end to the shape of the duct so that it can adjust easily in the hole of the duct. The inner piece has a length of 0.625 *in* and a diameter of 0.875 *in* at the surface, while the diameter at the step is 0.75 *in*. The outer piece has a length of 0.25 *in* and a diameter of 0.875 *in*.

A.4 The Lower End

The lower end of the UAV has flaps, servos to control flaps, a tail mount, a battery, a sonar, and another Electronic Speed Controller (ESC). The flaps are of two different kinds as shown in Fig.A.9 and Fig.A.10. Both flaps have a thickness of 0.125 *in*, are made of fibre glass, and are attached to each other using ordinary plastic hinges. The upper flap has a small cut at the lower left so that the tail mount can rest on it. The lower flap is the flap that has the job of controlling the outgoing direction of airflow and is designed to tilt a maximum of 30^o in either direction. There are three sets of flaps on each of the four sides, therefore, the total number of flaps is twelve. There are small flanges attached to one of the flaps on each side and a short mechanical linkage is used to connect the brackets to the servo that controls the angle of the ailerons.

It is attached to the duct at the other end using a plastic bracket shown in Fig.A.11. This bracket is also made out of plastic and holds the servo motor along with the flaps.

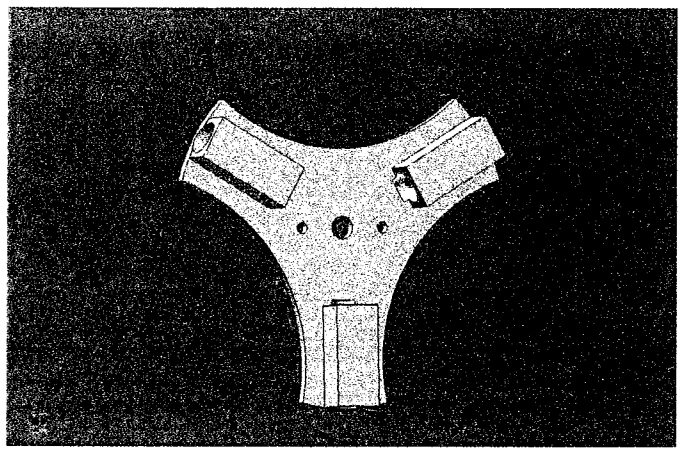

Figure A.6: The bottom motor mounting plate

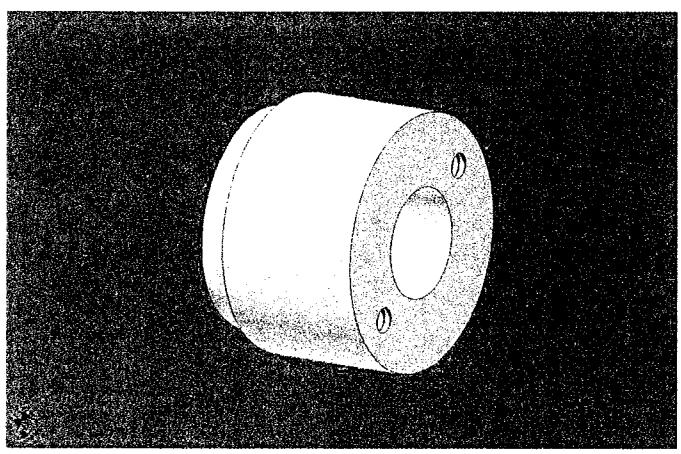

Figure A.7: Plastic bracket that goes inside the duct

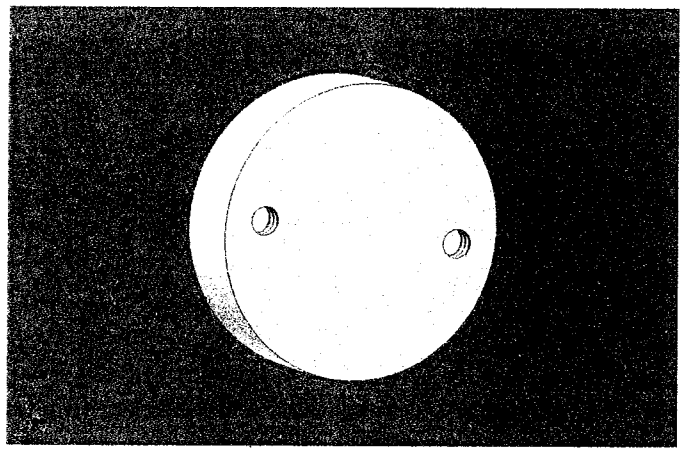

Figure A.8: Plastic bracket that goes outside the duct

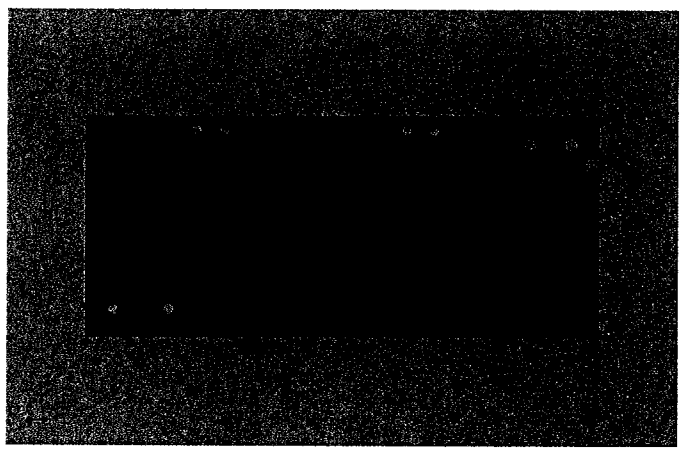

Figure A.9: The upper flap

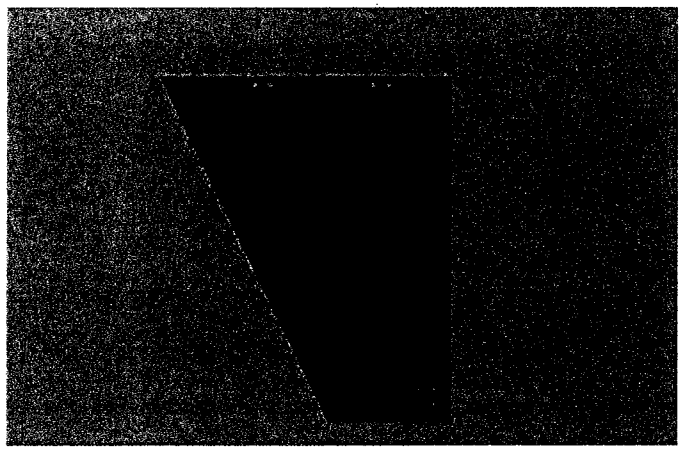

Figure A.10: The lower flap

Figure A.11: The bracket used to attach flaps

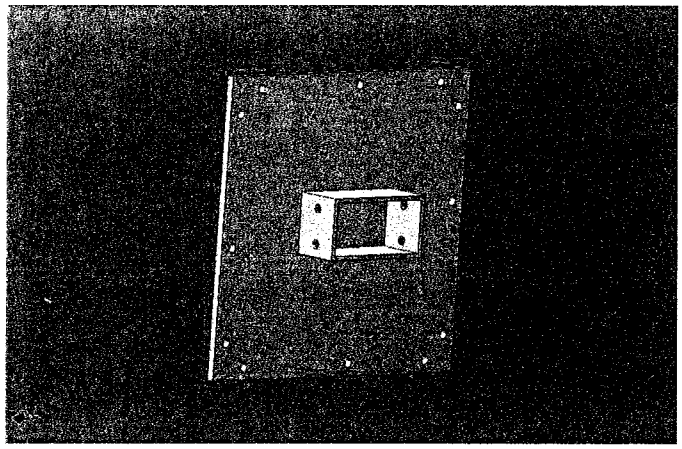

Figure A.12: The Tail Mount

A.5 The Tail

The tail consists of a tail mount and a small battery holder that holds a battery, a sonar, and an ESC. Both the tail mount and the battery holder are fabricated using light weight aluminum plate 0.125 *in* thick. The battery holder has two parts, one upper and one lower, that are attached together using nuts and bolts. The upper part of tail mount is fastened to the tail mount using screws. The tail mount is shown in Fig.A.12, while the battery holder is shown in Fig.A.13 and Fig.A.14.

A.6 The Landing Ring

The landing ring is attached to the UAV using landing-ring-brackets and landing-ring-shoes. The landing ring is a simple PVC tube having a diameter of 26.1 *in*, and a thickness of 0.25 *in*. Both brackets and shoes are made using lightweight plastic. The bracket is shown in Fig.A.15. The landing-ring-shoe consists of two parts, one upper, shown in Fig.A.16, and one lower, shown in Fig.A.17. Both parts of the shoe are fastened together using screws. The landing-ring is connected to the main body using carbon fiber tubes that can be inserted in the bracket at one end and in the shoe at the other end.

A.7 The Overall UAV Model

The overall UAV model is shown in Fig.A.18 and the cross-sectional view is shown in Fig.A.19

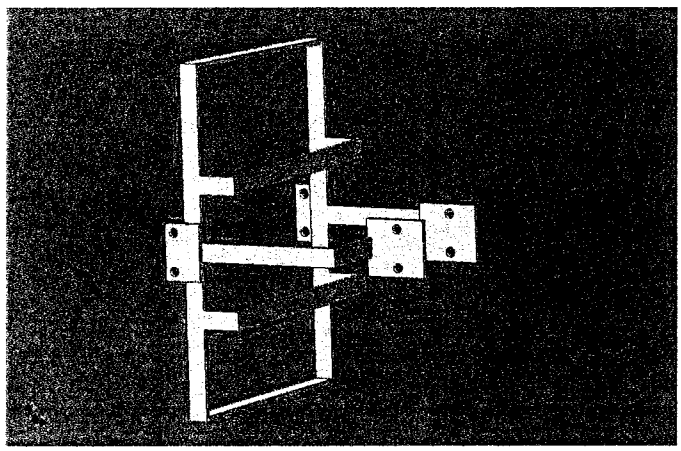

Figure A.13: The upper part of battery mount

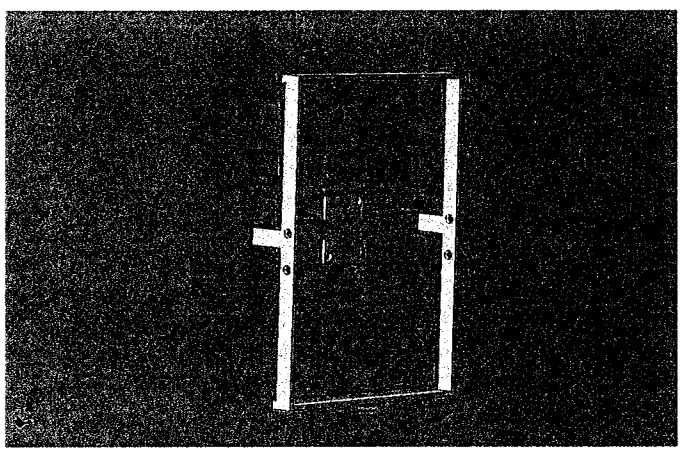

Figure A.14: The lower part of battery mount

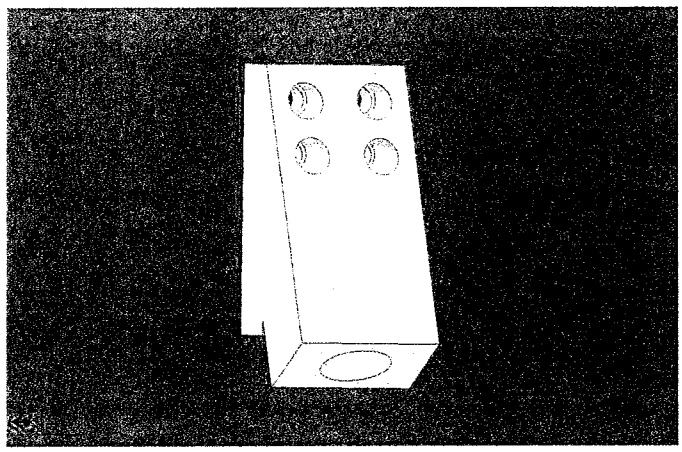

Figure A.15: The landing leg bracket

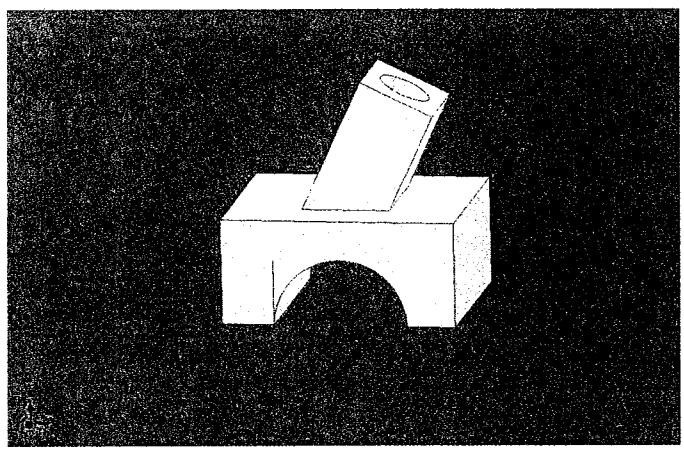

Figure A.16: The upper part of landing ring shoe

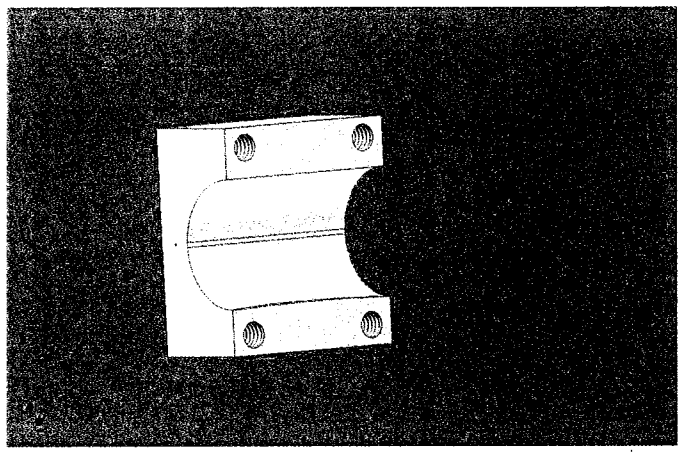

Figure A.17: The lower part of landing ring shoe

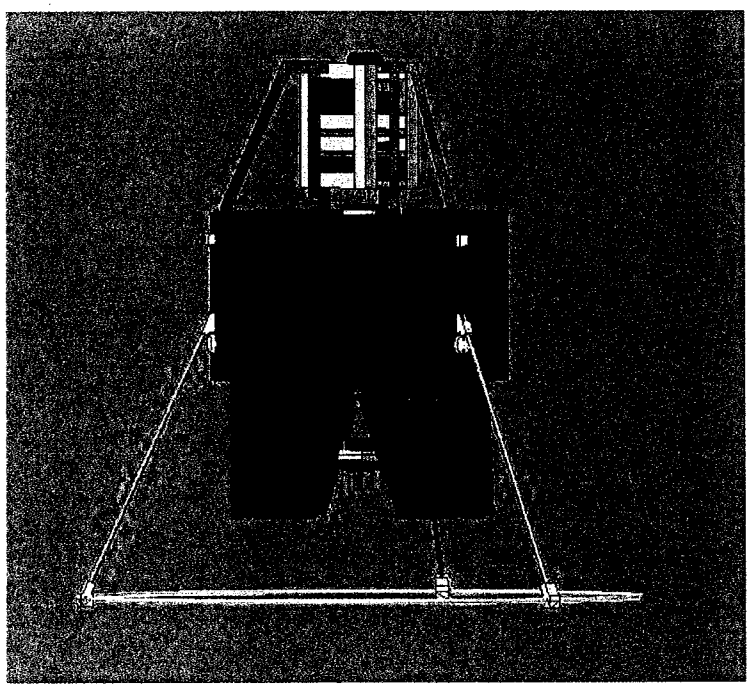

Figure A.18: The complete UAV model

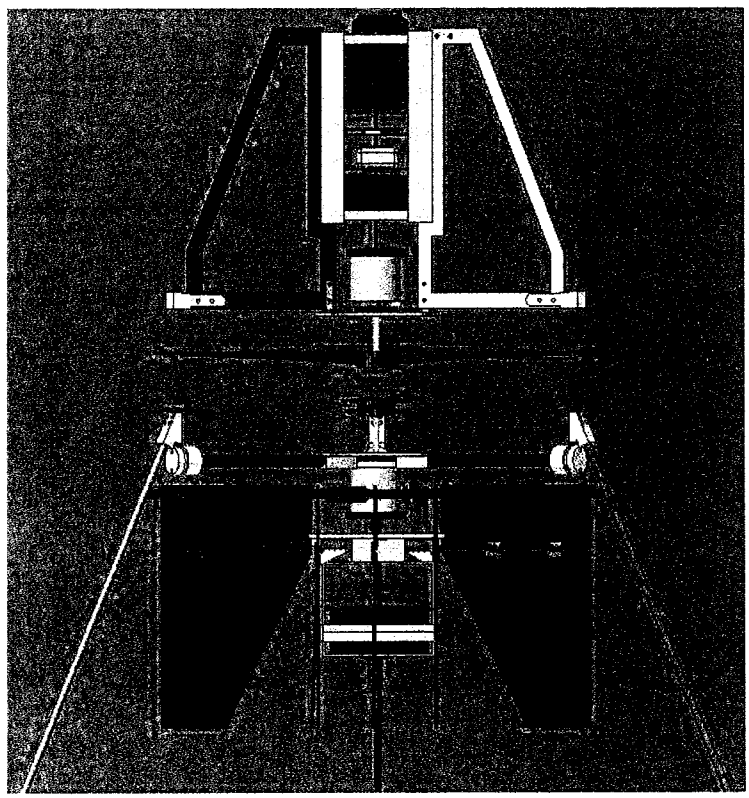

Figure A.19: The cross-sectional view of complete UAV model

Bibliography

[1] B.Thuilot, P. Martinet, L.Cordesses, and J.Gallice, "Position based visual servoing: Keeping the object in the field of vision" *Proceedings of the 2002 IEEE Conference on Robotics and Automation*, pp. 1624-1629, May 2002.

[2] F. Chaumette, and S. Hutchinson, "Visual Servo Control, Part I: Basic Approaches" *IEEE Robotics and Automation Magazine*, Vol. 13, No. 4, pp. 82-90, Dec 2006.

[3] L. Weiss, A. Sanderson, and C. Neuman, "Dynamic Sensor-Based Control of Robots with Visual Feedback" *IEEE Journal of Robotics and Automation*, Vol. RA-3, No. 5, pp. 404-417, Oct 1987.

[4] P. I. Corke, "Visual Control of Robot Manipulators - A review" *CSIRO Division of Manufacturing Technology*, Preston, Victoria, Australia, 1993.

[5] A.C. Sanderson, and L.E. Weiss, "Image-based Visual Servo Control of Robots" *26th Annual SPIE Technical Symposium*, Aug 1982.

[6] N. Guenard, T. Hamel, and R. Mahony, "A Practical Visual Servo Control for an Unmanned Aerial Vehicle" *IEEE Transactions on Robotics*, Vol. 24, No. 2, pp. 331-340, Apr 2008.

[7] B. Espiau, F. Chaumette, and P. Rives, "A New Approach to Visual Servoing Robotics" *IEEE Transactions on Robotics and Automation*, Vol. 8, No. 3, pp. 313-326, June 1992.

[8] T. Hamel, and R. Mahony, "Visual servoing of an under-actuated dynamic rigid-body system: an image-based approach" *IEEE Transactions on Robotics and Automation*, Vol. 18, No. 2, pp. 187-198, Apr 2002.

[9] X. Shen,and M. Chen, "A Fuzzy Self-Tuning Algorithm for Depth-Axis Control in Image-Based Visual Servo Control" *Fourth International Conference on Fuzzy Systems and Knowledge Discovery*, Vol. 4, pp. 149-153, Aug 2007.

[10] M. Iwatsuki, and N. Okiyama, "A new formulation of visual servoing based on cylindrical coordinate system" *IEEE Transactions on Robotics and Automation*, Vol. 21, No. 2, pp. 266-273, Apr 2005.

[11] F. Chaumette, and S. Hutchinson, "Visual Servo Control, Part II: Advanced Approaches " *IEEE Robotics and Automation Magazine*, Vol. 14, No. 1, pp. 109-118, Mar 2007.

[12] P.C. Hughes, *"Spacecraft Attitude Dynamics"*, Dover Publications Inc., 2004.

[13] R.M. Murray, Z. Li, S.S. Sastry, *"A Mathematical Introduction to Robotic Manipulation"*, CRC Press, 1994.

[14] M.D. Shuster, "A Survey of Attitude Representations", *The Journal of Astronomical Sciences*, Vol. 41, No. 4, Dec. 1993, pp. 439-517.

[15] M.W. Spong, and M. Vidyasagar, *Robot Dynamics and Control'*, 2006.

[16] J. Katz, and A. Plotkin, *"Low-Speed Aerodynamics"*, 2nd Edition, Cambridge University Press, 2000.

[17] A. Roberts, and A. Tayebi, "Design, Modelling, and Control of a Ducted Fan VTOL UAV", *M.Sc. Thesis*, Lakehead University, Canada.

[18] Shapiro, L. G., and G. C. Stockman. *Computer vision.* Upper Saddle River: Prentice Hall, 2001.

[19] C. F. Olson, "Efficient Pose Clustering Using a Randomized Algorithm" *International Journal of Computer Vision*, Vol. 23, pp. 131-147, 1997.

[20] C. F. Olson, "Pose Clustering Guided by Short Interpretation Trees " *Proceedings of the 17th International Conference on Pattern Recognition*, Vol. 2, pp. 149-152, 2004.

[21] W. Wang, J. Gu, and T. C. Henderson, "A Pipelined Architecture for Parallel Image Relaxation Operations" *IEEE Transactions on Circuits and Systems*, Vol. 34 No. 11, pp. 1375 -1384, 1987.

[22] R. R. Beichel. Digital Image Processing. Course Website. College Of Computer Systems Support, University of Iowa. 16 Oct 2007.

$< www.icaen.uiowa.edu/\ dip/LECTURE/Understanding5.html >$

[23] A. Broadhurst, T.W. Drummond and R. Cipolla, "A Probabilistic Framework for Space Carving" *Eighth IEEE Conference on Computer Vision*, Vol. 1, pp. 388-393, 2001.

[24] M. Vincze, and G. D. Hager, *"Robust Vision for Vision-Based Control of Motion"*, Piscatawa: IEEE Press, 2000.

[25] T. Hamel, and R. Mahony, "Visual Servoing of an Under-Actuated Dynamic Rigid Body System: An Image-Based Approach" *IEEE Transactions on Robotics and Automation*, Vol. 18 No. 2, pp. 187-198, 2002.

[26] N. Guenard, T. Hamel, and R. Mahony, "A practical visual servo control for a unmanned aerial vehicle" *IEEE International Conference on Robotics and Automation*, pp. 1342 - 1348, 2007.

[27] R. Olfati-Saber, "Global configuration stabilization for the VTOL Aircraft with Strong Input Coupling" *IEEE Transactions on Automatic Control*, Vol. 47 No. 11, pp. 1949 - 1952, 2002.

[28] J. Pflimlin, P. Soueres, and T. Hamel, "Hovering flight stabilization in wind gusts for ducted fan UAV" *43rd IEEE Conference on Decision and Control*, pp. 3491 - 3496, 2004.

[29] L. Lipera, "The Micro Craft iSTAR Micro Air Vehicle: Control System Design and Testing", *American Helicopter Society 57th Annual Forum*, 2001

[30] A. Tayebi, S. McGilvray, A. Roberts, and M. Moallem, "Attitude estimation and stabilization of a rigid body using low-cost sensors", *In proceedings of the 46th IEEE conference on decision and control*, New Orleans, LA, USA, 2007.

[31] J.G. Leishman, *Principles of Helicopter Dynamics*, Cambridge University Press, 2000.

[32] R.W. Prouty, *Helicopter Performance, Stability, and Control*, Robert E. Krieger Publishing Company Inc., 1986.

[33] B. Herisse, Francois-Xavier Russotto, Tarek Hamel, and Robert Mahon, "Hovering flight and vertical landing control of a VTOL Unmanned Aerial Vehicle using Optical Flow" *IEEE/RSJ International Conference on Intelligent Robotics and Systems*, pp. 801 -808, 2008.

[34] F. Le Bras, T. Hamel, and R. Mahony, "Visual Servoing of a VTOL Vehicle using Virtual States", *Proceedings of the 46th IEEE Conference on Decision and Control*, New Orleans, LA, USA pp. 6442 - 6447, 2007.

[35] T. Cheviron, T. Hamel, and G. Baldwin, "Robust Nonlinear Fusion of Inertial and Visual Data for position, velocity and attitude estimation of UAV", *IEEE International Conference on Robotics and Automation*, Roma, Italy, pp. 2010 - 2016, 2007

[36] P. van Blyenburgh, "UAVs: an Overview", *Air and Space Europe*, Vol. 1 No. 5/6, pp. 43-47, 1999.

[37] P. Spanoudakis, N. C. Tsourveloudis, and K. P. Valavanis, "Design Specifications for Unmanned VTOL", *IEEE Inbrnstlonal Conference am Robotla L Automation*, New Orleans, LA, USA, pp. 3616-3622, 2004.