Quaternion Based Pointing Algorithm for Two-Axis Gimbal of Micro Aerial Vehicles

Meenu Selvarajan, C M Ananda

Abstract— **With the increased use and demand of Unmanned Aerial Vehicle (UAV) and Micro Aerial Vehicle (MAV) for military, societal, security, coast guards, surveillance, entertainment etc., MAVs demand precision payload with well stabilized and controllable gimbal system. The fidelity of the data is directly dependent on the degree of control of the gimbal. Hence an efficient gimbal controller plays an important role in the success of the UAV/MAV mission. This paper uses unit quaternions for rotations and derives an algorithm to point a camera, mounted with a two-axis (roll-pitch) gimbal on a Micro Aerial Vehicle (MAV), towards a target at a known location on the ground. The use of unit quaternions makes the algorithm more computationally efficient and free of mathematical singularities. The inputs required are the GPS coordinates of the MAV and the target along with the attitude information of the MAV measured using Inertial Measurement Unit (IMU). The algorithm calculates the gimbal roll pitch angles required to point the camera at the target.**

Keywords— Unmanned aerial vehicles, Quaternions, Gimbal, PID.

I. Introduction

Unmanned Aerial Vehicles (UAVs) are increasingly finding use in numerous military applications all over the globe, primarily for Intelligence, Surveillance, and Reconnaissance (ISR) tasks [1]. Micro Aerial Vehicle (MAV) refers to class of fixed wing UAV with a smaller wing span. These are typically battery powered, hand launched and belly landed vehicles. Their small size enables them to be used in several reconnaissance operations which may not be possible with a large sized UAV. Of interest in this paper is one such surveillance system which consists of a camera mounted via a two-axis gimbal on a fixed wing MAV. The gimbal allows the camera to be rotated along two axes. The requirement is to point the camera at a known location on the ground or on air in the field of view of the MAV.

An excellent explanation of the working of Unmanned Aerial Systems is given in [1]. J.M. Hilkert in [2] explains the various technologies used in inertially stabilized platforms with gimbals that point sensors towards a target. Semke, et al. [3] [4] does a thorough kinematic analysis of the three-axis gimbal pointing problem and derives an analytical solution for the angles at which the gimbal has to be oriented using Euler angles. Euler angle approach though very popular, has mathematical singularities at certain angles. For example, in the ψ-θ-φ Euler Angle sequence, there arises a singularity when θ = 90°. At this point, the values of ψ and φ are not defined, resulting in computational instabilities. This phenomenon is commonly referred to as gimbal lock [1] and can be overcome with the use of quaternions. The basics of quaternions and how they can be used for aerospace rotations are very well explained in [5]. A quaternion based gimbal attitude control algorithm is presented by Weiss in [6]. Methodology

Meenu Selvarajan : Post Graduate Student, Department of Mathematical and Computational Sciences, National Institute of Technology, Surathkal, Mangalore, Karnataka, India(selvarajan.meenu@gmail.com)

C M Ananda, Head: Department of Aerospace, Electronics and Systems Division, CSIR-National Aerospace Laboratories, Bangalore, Karnataka, India 560017,(ananda.cm@nal.res.in)

II. MATHEMATICAL PRELIMINARIES: QUATERNIONS [5]

Quaternions are hyper complex numbers of rank four discovered by William Rowan Hamilton in the year 1843. The general form of a quaternion is, $q = q_0 + iq_1 + jq_2 + kq_3$ Where q_0, q_1, q_2, q_3 are real numbers. q_0 is the scalar part of the quaternion and $\bar{q} = iq_1 + jq_2 + kq_3$ is called the vector part and,

$$i^2 = j^2 = k^2 = ijk = -1$$
$$ij = k = -ji$$
$$jk = i = -kj$$
$$ki = j = -ik$$

The *complex conjugate* of q is $q^* = q_0 - iq_1 - jq_2 - kq_3$ and the *norm* is defined as

$$N(q) = \sqrt{q^*q} = \sqrt{q_0^2 + q_1^2 + q_2^2 + q_3^2}$$

If the norm is 1, the quaternion is called a unit quaternion, which can be represented by

$$q = q_0 + \bar{q} = \cos\theta + u\sin\theta$$

Where, u is the unit vector along the direction of $\bar{q}$.

Quaternions along with the operations addition and multiplication form a ring, which means quaternion multiplication is not commutative. It is this property of quaternions which makes it suitable for representing rotations. A vector in R^3 can be considered as a quaternion whose scalar part is 0 (pure quaternion).For any unit quaternion q and any vector $v \in R^3$ (represented as a unit quaternion), the action of the operator

$$L_q^*(v) = q^*vq$$

can be interpreted geometrically as a rotation of the coordinate frame with respect to the vector *v* through an angle 2θ about q as the axis[5].

III. KINEMATIC ANALYSIS

Three coordinate systems are used to describe the position and orientation of the MAV and the camera with respect to each other and with respect to earth: inertial frame, body (MAV) frame (Fig. 1), gimbal (or camera) frame. These frames are related by rotations. A flat non-rotating earth is assumed in the analysis. Inertial coordinate system: x axis is along the North direction, y axis along the East, z axis in the vertically downwards direction with the origin at the defined home location. This is sometimes referred to as North-East-

978-1-5090-0774-5/16/$31.00 © 2016 IEEE

Down (NED) reference frame [1]. The body frame (Fig. 1): x axis points through the nose of the MAV, y axis points to the right of the x-axis and z axis points down through the bottom the craft, perpendicular to the x-y plane and satisfying the right hand rule [7]. Rotation of the body about x axis is known as roll (φ), about y axis is known as pitch (θ), about z axis is known as yaw (ψ). The body frame is initially aligned with the inertial frame.

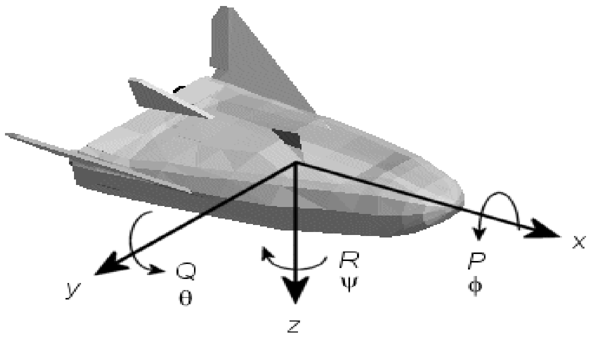

Fig. 1 Body Frame [7]

For the sake of simplicity the origins of the body frame and gimbal frame are assumed to be at the same point (center of mass of the MAV). Initially the gimbal axes coincide with the body frame, with the *z* axis assumed to be the optical axis of the camera attached to the gimbal. As the gimbal movement is coupled to the body rotations, the gimbal axes undergo the same sequence of rotations as the body frame. The aim is to align the z-axis (optical axis) of the gimbal along the line of sight irrespective of the aircraft rotations [4].

Let the position of the MAV in inertial plane be P_{mav} and the position of the target location in inertial plane be P_{target} [4].

The line of sight is obtained as,

$$\begin{aligned} l &= P_{target} - P_{mav} \\ &= xi + yj + zk \end{aligned} \tag{1}$$

Consider the *yaw-pitch-roll: ψ-θ-φ* (aerospace sequence) rotation of the MAV. The unit quaternions for the rotations are

$$q_{yaw} = \cos\frac{\psi}{2} + k\sin\frac{\psi}{2} \tag{2}$$

$$q_{pitch} = \cos\frac{\theta}{2} + j\sin\frac{\theta}{2} \tag{3}$$

$$q_{roll} = \cos\frac{\varphi}{2} + i\sin\frac{\varphi}{2} \tag{4}$$

To transform the line of sight from inertial to body frame the operator,

$$L_{q^*_{body}}(l) = q^*_{body}\, l\, q_{body} \tag{5}$$

is used, where,

$$\begin{aligned} q_{body} &= q_{yaw} q_{pitch} q_{roll} \\ &= (\cos\frac{\psi}{2}\cos\frac{\theta}{2}\cos\frac{\varphi}{2} + \sin\frac{\psi}{2}\sin\frac{\theta}{2}\sin\frac{\varphi}{2}) \\ &\quad + i(\cos\frac{\psi}{2}\cos\frac{\theta}{2}\sin\frac{\varphi}{2} - \sin\frac{\psi}{2}\sin\frac{\theta}{2}\cos\frac{\varphi}{2}) \\ &\quad + j(\cos\frac{\psi}{2}\sin\frac{\theta}{2}\cos\frac{\varphi}{2} + \sin\frac{\psi}{2}\cos\frac{\theta}{2}\sin\frac{\varphi}{2}) \\ &\quad - k(\cos\frac{\psi}{2}\sin\frac{\theta}{2}\sin\frac{\varphi}{2} + \sin\frac{\psi}{2}\cos\frac{\theta}{2}\cos\frac{\varphi}{2}) \end{aligned} \tag{6}$$

After the transformation using Eq. (5), line of sight in the body frame is given by,

$$\begin{aligned} l_b = {} & i(-z\sin\theta + x\cos\psi\cos\theta + y\sin\psi\cos\theta) \\ & + j(y\cos\psi\cos\varphi - x\sin\psi\cos\varphi + z\cos\theta\sin\phi \\ & \quad + x\cos\psi\sin\theta\sin\varphi + y\sin\psi\sin\theta\sin\varphi) \\ & + k(z\cos\theta\cos\varphi - y\cos\psi\sin\varphi + x\sin\psi\sin\varphi \\ & \quad + x\cos\psi\sin\theta\cos\varphi + y\sin\psi\sin\theta\cos\varphi) \end{aligned} \tag{7}$$

At this point, the problem reduces to finding a set of gimbal roll pitch rotations which will align the *z* axis of the gimbal frame with the line of sight given by Eq. (7). Let the required rotations be θ_{roll} and θ_{pitch} in that sequence. The corresponding quaternions are,

$$q_{\theta_{roll}} = \cos\frac{\theta_{roll}}{2} + i\sin\frac{\theta_{roll}}{2} \tag{8}$$

$$q_{\theta_{pitch}} = \cos\frac{\theta_{pitch}}{2} + j\sin\frac{\theta_{pitch}}{2} \tag{9}$$

The total rotation quaternion for the gimbal frame is given by,

$$\begin{aligned} q_{gimbal} &= q_{\theta_{roll}} q_{\theta_{pitch}} \\ &= \cos\frac{\theta_{roll}}{2}\cos\frac{\theta_{pitch}}{2} \\ &\quad + i(\sin\frac{\theta_{roll}}{2}\cos\frac{\theta_{pitch}}{2}) \\ &\quad + j\left(\cos\frac{\theta_{roll}}{2}\sin\frac{\theta_{pitch}}{2}\right) \\ &\quad + k(\sin\frac{\theta_{roll}}{2}\sin\frac{\theta_{pitch}}{2}) \end{aligned} \tag{10}$$

To transform the line of sight l_b obtained in Eq. (7) to gimbal frame, the following operation is performed.

$$L_{q^*_{gimbal}} = q^*_{gimbal}\, l_b\, q_{gimbal} \tag{11}$$

This gives the line of sight in gimbal frame as,

$$\begin{aligned} l_{final} = {} & i(\sin\theta_{pitch}\,(\sin\theta_{roll}\,(\sin\varphi\,(\sin\theta\,(x\cos\psi \\ & \quad + y\sin\psi) + z\cos\theta) + \cos\phi\,(y\cos\psi - x\sin\psi)) \\ & \quad - \cos\theta_{roll}\,(\sin\theta\cos\varphi\,(x\cos\psi + y\sin\psi)) \\ & \quad + \sin\varphi\,(x\sin\psi - y\cos\psi) + z\cos\theta\cos\varphi)) \\ & \quad + \cos\theta_{pitch}\,(\cos\theta\,(x\cos\psi + y\sin\psi) - z\sin\theta)) \\ & + j(\sin\theta_{roll}\,(\sin\theta\cos\varphi\,(x\cos\psi + y\sin\psi) \\ & \quad + \sin\varphi\,(x\sin\psi - y\cos\psi) + z\cos\theta\cos\varphi) \\ & \quad + \cos\theta_{roll}\,(\sin\varphi\,(\sin\theta\,(x\cos\psi + y\sin\psi) \\ & \quad + z\cos\theta) + \cos\varphi\,(y\cos\psi - x\sin\psi))) \\ & + k(\cos\theta_{roll}\,\cos\theta_{pitch}\,(\sin\theta\cos\varphi\,(x\cos\psi \\ & \quad + y\sin\psi) + \sin\varphi\,(x\sin\psi - y\cos\psi) \\ & \quad + z\cos\theta\cos\varphi) \\ & \quad - \sin\theta_{roll}\,\cos\theta_{pitch}\,(\sin\varphi\,(\sin\theta\,(x\cos\psi \\ & \quad + y\sin\psi) + z\cos\theta) + \cos\varphi\,(y\cos\psi - x\sin\psi)) \\ & + \sin\theta_{pitch}\,(\cos\theta\,(x\cos\psi + y\sin\psi) - z\sin\theta)) \end{aligned} \tag{12}$$

The above vector should lie along the z axis of the gimbal which is the assumed optical axis of the camera. As a result, there will not be any components along the *x* and *y* axes of the gimbal. This condition gives rise to the following two non-linear equations

$$\sin\theta_{pitch}\,(\sin\theta_{roll}\,(\sin\varphi\,(\sin\theta\,(x\cos\psi + y\sin\psi) + z\cos\theta) + \cos\phi\,(y\cos\psi - x\sin\psi)) - \cos\theta_{roll}\,(\sin\theta\cos\varphi(x\cos\psi + y\sin\psi)) + \sin\varphi\,(x\sin\psi - y\cos\psi) + z\cos\theta\cos\varphi)) + \cos\theta_{pitch}\,(\cos\theta\,(x\cos\psi + y\sin\psi) - z\sin\theta)) = 0 \quad (13)$$

$$\sin\theta_{roll}\,(\sin\theta\cos\varphi\,(x\cos\psi + y\sin\psi) + \sin\varphi\,(x\sin\psi - y\cos\psi + z\cos\theta\cos\varphi) + \cos\theta_{roll}\,(\sin\varphi\,(\sin\theta\,(x\cos\psi + y\sin\psi)) + z\cos\theta) + \cos\varphi\,(y\cos\psi - x\sin\psi))) = 0 \quad (14)$$

Solving equations (13) and (14) using Mathematica, the values of θ_{roll} and θ_{pitch} can be obtained as

$$\theta_{roll} = \tan^{-1}\frac{A_r}{B_r} \quad (15)$$

where,

$$A_r = -y\cos\psi\cos\varphi + x\sin\psi\cos\varphi - z\cos\theta\sin\varphi - x\cos\psi\sin\theta\sin\varphi - y\sin\psi\sin\theta\sin\varphi$$

$$B_r = z\cos\theta\cos\varphi - y\cos\psi\sin\varphi + x\sin\psi\sin\varphi + x\cos\psi\sin\theta\cos\varphi + y\sin\psi\sin\theta\cos\varphi$$

and,

$$\theta_{pitch} = \tan^{-1}\frac{A_p}{B_p} \quad (16)$$

where,

$$A_p = -z\sin\theta + x\cos\psi\cos\theta + y\sin\psi\cos\theta$$

$$B_p = z^2\cos^2\theta + y^2\cos^2\psi + x^2\sin^2\theta\cos^2\psi + yz\sin\psi\sin 2\theta + x^2\sin^2\psi + y^2\sin^2\theta\sin^2\psi + xz\cos\psi\sin 2\theta - 2xy\sin\psi\cos\psi + xy\sin^2\theta\sin 2\psi$$

Equations (15) and (16) give the angles by which the gimbal has to be rotated in order to point the camera at the desired target. Considering the case when the target is directly below the MAV, line of sight vector is $0i + 0j + 1k$. Solving equations (15) and (16) then gives $\theta_{roll} = -\varphi$ and $\theta_{pitch} = -\theta$. This agrees with the intuitive solution that the gimbal has to perform the inverse of the rotations performed by the MAV to restore it to initial configuration.

IV. DC MOTOR CONTROL

The actuators used in the gimbal are small DC motors. Control of DC motors is a well-researched and understood topic. Figure 2 shows the block diagram model of a DC motor.

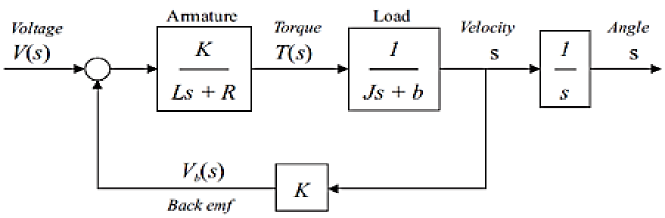

Fig.2 DC Motor Model [8]

The transfer function of the DC motor [8] is given by

$$\frac{\theta(s)}{V(s)} = \frac{K}{s[(Ls + R)(Js + B) + K^2]} \quad (17)$$

where, K is the back electromotive force constant in *Nm/A*, L is the electric inductance in *Henry, R* is the electric resistance in *Ohm, J* is the moment of inertia of the rotor in $kg\ m^2$, B is the damping (friction) of the mechanical system in *Nms,* $\theta(s)$ is the angular position in *radians* [9].

This paper makes use of a Proportional Integral Derivative (PID) controller [10] to control the position of the motor. Even after the development of better controllers, PID still remains the industry favourite because of its simplicity and ease of implementation. The output of a PID controller is given by,

$$pid_{out} = k_p e(t) + ki\int_0^t e(x)\,dx + k_d\frac{de(t)}{dt} \quad (18)$$

V. SIMULATION RESULTS

For initial testing of equations (15) and (16), the MAV and target locations were assumed to be constant at $(0,0,0)$ and $(\frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}})$ respectively (values assumed so that the line of sight is a unit vector). Sample flight data containing the attitude information of the MAV, i.e. yaw(ψ), pitch(θ), roll(φ) were used to calculate the values of θ_{roll} and θ_{pitch}. Figure 3 shows one such set of test data. Figures 4 and 5 shows the calculated roll and pitch angles for the data sample in Fig. 3.

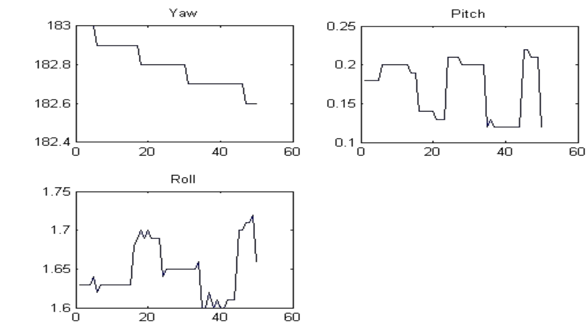

Fig.3 Test Data (Attitude)

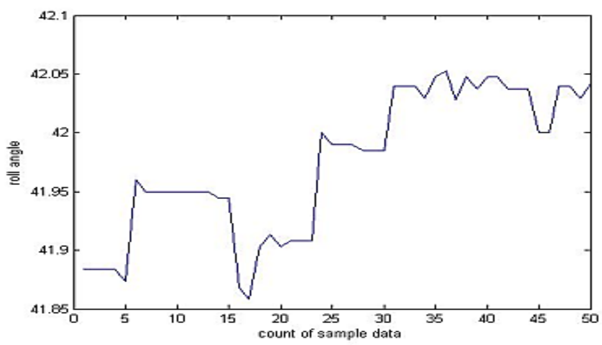

Fig.4 Roll Angles Calculated using Eq. (15)

A MATLAB script file was created to simulate the rotations of the line of sight vector. Applying the yaw-pitch-roll rotations as in Fig. 3 followed by the gimbal roll-pitch rotations as in figures 4 and 5 aligns the line of sight vector along the *z* axis which is the assumed optical axis (Fig. 7), thus verifying equations (15) and (16). The actuator responses for the commanded angles were checked using a Simulink model of the DC motor as per Fig. 2 and a PID controller.

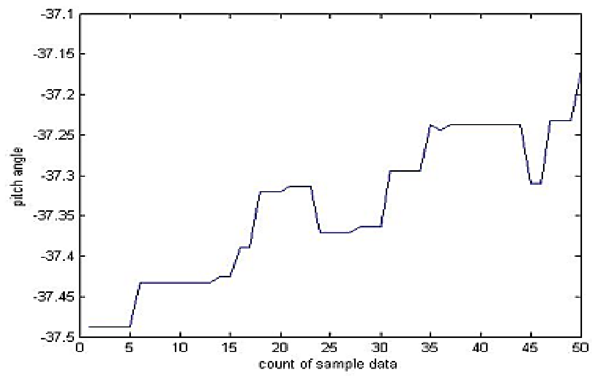

Fig.5 Pitch Angles Calculated using Eq. (16)

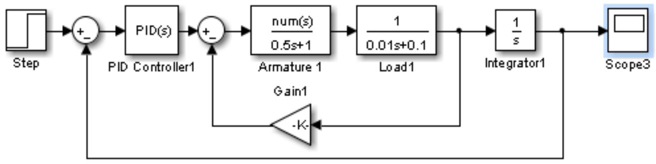

Fig.6 Actuator Model

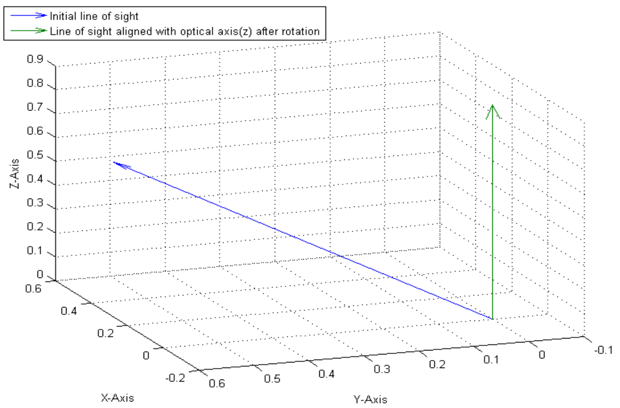

Fig.7 Rotation Simulation

The following parameters were assumed for the model [8]: *J = 0:01* $Kg\ m^2$, *K=0.01 Nm/A*, *B = 0.1 Nms*, *R = 1 Ohm*, *L = 0.5 Henry*. Fig. 6 shows the Simulink model with PID Controller. Tuning of the controller was done with the auto tune feature of MATLAB. The response of the motor for a typical test angle of $-30°$ is shown in Fig. 8

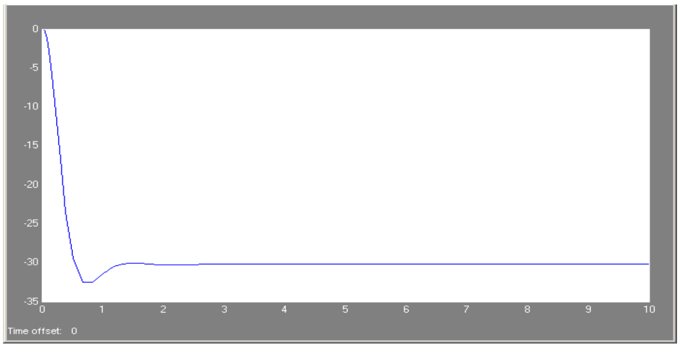

Fig.8 Response of the simulated actuator for an angle of $-30°$

VI. EXPERIMENTAL RESULTS

The experimental setup as in Fig. 9 consists of a two axis gimbal with TowerPro SG-90 micro servos as actuators. These servos, weighing around 15 g, can rotate approximately $180°$ ($90°$ in each direction) with Pulse Width Modulation (PWM) pulses in the range of 1 ms-2 ms, at a voltage range of 4.8-6 V producing a torque of 2.5 kg-cm. The controller used is a Cypress Semiconductor Programmable System on Chip (PSoC) 5 board which uses a ARM Cortex-M3 processor. The feedback for the motor position is taken with the help of a Motion Processing Unit (MPU) 9150 sensor.

In the first set of experiments, a special case was considered where the line of sight as in Eq. (1) is $0i + 0j + 1k$ (perpendicular to the ground) and the rotation involves only pitching ($\psi = 0, \varphi = 0$). This results in $\theta_{pitch} = -\theta$, i.e. the pitch axis actuator has to be rotated in the opposite direction of the MAV pitching, by the same angle. For example when the MAV pitches by $30°$, the gimbal has to pitch by $-30°$ to maintain line of sight perpendicular to the ground.

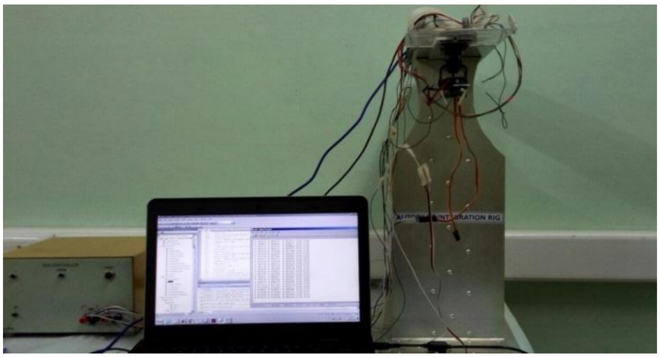

Fig.9. Experimental Setup

A discrete PID controller for the pitch axis servo using Eq. (19) was implemented

$$pid_{out} = k_p * eroor + k_i * errorsum + k\_d * errordifference$$

where,

$$errorsum = \sum error$$

$$errordifference = error - previouserror$$

The parameters k_p , k_i, k_d were found with trial and error method as $0.3, 0.005, 0.55$ respectively. Fig. 10 shows the response of the pitch-axis servo when the MAV pitching is $30°$.

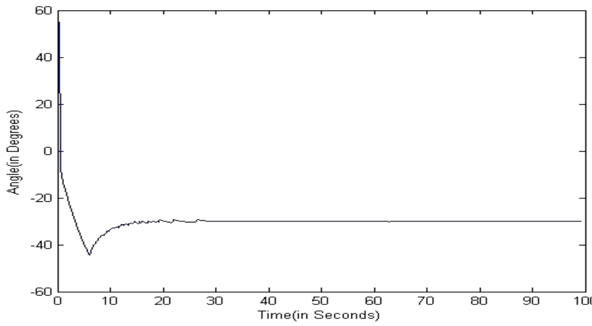

Fig.10 Response of the pitch axis servo

As can be seen by comparing Fig. 8 and Fig. 10, the actuator behaves in a similar manner as in the simulation. But the settling time in the experimental response is much greater than in the simulation which would not be acceptable in real time scenarios. This difference can be attributed to the delay in measuring the motor position using the sensor and also to the difference in the simulation parameters and the actual motor parameters and has to be reduced.

VII. CONCLUSION

Use of quaternions for rotations was studied and an analytical solution was derived for the gimbal pointing problem. The obtained solution was verified using MATLAB with sample flight data. Hardware implementation of a sample test case with pitch movement was done with a servo motor and PID controller. The complete algorithm has to be implemented and tested along with the vision system integrated with the hardware-in-the-loop simulation.

ACKNOWLEDGMENT

The authors would like to thank the team at the National Aerospace Laboratories - Microelectronics and Systems Laboratory (NAL-MSL) for their support and contributions towards the project. The authors would also like to thank Professor A. Kandasamy, Department of Mathematical and Computational Sciences, NITK Surathkal, for his guidance and support.

REFERENCES

[1] Randal.E Beard and Timothy W. McLain, "*Small Unmanned Aircraft: Theory and Practice*", Princeton and Oxford: Princeton University Press, 2012.

[2] J.M. Hilkert, "*Inertially stabilized platform technology Concepts and principles*", in *IEEE Control Systems*, vol. 28, no. 1, pp. 26-46, Feb. 2008

[3] Semke W., Ranganathan J. and Buisker M. , "*Active Gimbal Control for Surveillance using Small Unmanned Aircraft Systems*", Proceedings of the International Modal Analysis Conference(IMAC) XXVI: A conference and Exposition on Structural Dynamics, February 2008.

[4] J. Ranganathan, and W. Semke., "*Three-Axis Gimbal Surveillance Algorithms for Use in Small UAS*," Proceedings of the ASME International Mechanical Engineering Conference and Exposition, IMECE2008-67667, 2008

[5] Jack .B Kuipers , "*Quaternions and Rotation Sequences: A Primer with Applications to Orbits,Aerospace, and Virtual Reality*", Princeton, New Jersey: Princeton University Press, 1999.

[6] Weiss H., "*Quaternion-Based Rate/Attitude Tracking System with Application to Gimbal Attitude Control*", Journal of Guidance, Control, and Dynamics, Vol.16,No.4,1993.

[7] T. MathWorks, *"About aerospace coordinate systems,"* 1994.[Online].Available:http://in.mathworks.com/help/aeroblks/about-aerospace-coordinate-systems.html.

[8] R. Babuškaand S. Stramigioli, *Matlab and Simulink for modeling and control*,1999.[Online].Available: http://www.dcsc.tudelft.nl/~sc4070/transp/refresher.pdf.

[9] R. J. Rajesh and C. M. Ananda, "PSO tuned PID controller for controlling camera position in UAV using 2-axis gimbal," *Power and Advanced Control Engineering (ICPACE), 2015 International Conference on*, Bangalore, 2015, pp. 128-133.

[10] *"Control tutorials for MATLAB and Simulink introduction: PID controller design,"* [Online]. Available: http://ctms.engin.umich.edu/CTMS/index.php?example=Introduction&section=ControlPID