Robotic Eye Design for Fast Moving Stimulus in Dynamic Social Environment

By

Muhammad Nouman	Reg. no:02-3-1-029-2016
Muhammad Umair Aziz	Reg. no:02-3-1-033-2016
Muhammad Waqas	Reg. no:02-3-1-034-2016

Thesis submitted to the faculty of Engineering in partial fulfillment of requirements for the Degree of BS Electrical Engineering.

Department of Electrical Engineering,

Pakistan Institute of Engineering and Applied Sciences,

Nilore, Islamabad 45650, Pakistan

December 26, 2019

Contents

Abstract

In dynamic social environment, fast eye ball movement in robotic eye is much needed to recognize and track the fast moving stimulus. For tracking of fast moving stimulus, high decoupling ratio of camera sensor from the head skeleton is required. Two axis gimbal system is used to achieve high decoupling ratio. Mathematical modeling of gimbal system is done by considering the torque disturbances and dynamic unbalance. For efficient tracking and independent eye movement in the bumpy environment, stabilization control system is designed. Non-linear System equations are linearized using Jacobain matrix to achieve linear state space equations. The overall model is simulated on the Simulink with CAD model of the gimbal and proportional integral derivative (PID) controller is used for the system. Simulation results confirms the tracking of the stimulus object.

Chapter 1

Introduction

Robotics deals with design, constructions and use of computer system for controlling the robots. The field of robotics also connects with artificial intelligence and machine learning. Nowadays, humanoid robots are getting popularity which includes full human body design and construction. Swarm of robots are also used for surveillance and for carrying object that require more robots. Working in swarm of robots, fast eye movement is required in order to track different robots for collecting more information. As in human, eyes are the major organ which collects more than 80 percent of information from the environment. So robots use camera sensor as eye to acquire information about its surrounding environment.

In robotic eye, fast movement of eye ball is required to track the fast moving stimulus which is necessary in case of communication between swarm of robots. But in dynamic social environment we have effect of unstructured environment. Camera should be decoupled from the robot body and head for fast moving stimulus tracking. Just like human eye, which is capable of locking the object in the center of retina even when position of head changes.

1.1 Gimbal System

Gimbal system can decouple the camera sensor from its base if we place the camera in the inner channel. Gimbal is a support that allows the rotation of an object about single axis and it provides decoupling of camera from its base on which it is mounted.

Gimbal system can have two or three axis depending upon the requirement of freedom we want. A three axis gimbal is shown in figure 1.1. In three axis gimbal, we have yaw, pitch and roll axis rotation. So anything in the inner channel gimbal can be decoupled from the support on which it is mounted. Due to high decoupling ratio of gimbal system, tracking fast stimulus in the dynamic environment using simple camera senor can be achieved with simple control design techniques.

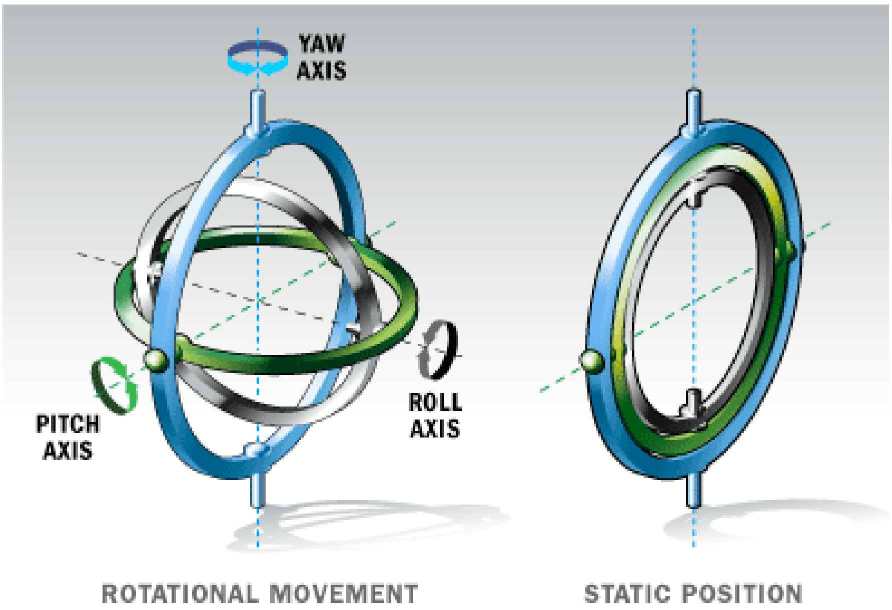

Figure 1.1: Three axis gimbal system [?]

1.2 Objective

Objective of robotic eye is to track the fast moving stimulus in dynamic environment without head movement as in human eye. So gimbal system has to modeled with gyro sensors and camera on the inner axis so that we can get the maximum decoupling ratio. For tracking, control algorithms and AI based vision control schemes can be used.

1.3 General Control of Gimbal

Newton law of rotational body is used to design control system of gimbal. According to the newton second law, if more than one torque acts on the body then the net torque is the sum of all the torques due to which body rotates. If the net torque is zero than the body does not rotate.

In dynamic environment, there is always some torque acts on the body, so we have to make net toque to zero to make body static. In gimbal system, disturbance torque is applied on one the gimbal axis and also on the base body due to disturbance in the environment. We insert motors to provide proportional torque which counterbalance the disturbance torque. In this way, line of sight(LOS) is unaffected by the base body's motion.

1.4 Thesis Outline

In this thesis, basic concept of robotic eye and motivation behind using gimbal system is discussed in first chapter. Second Chapter presents the literature review of robotic eye and gimbal system.

Different approaches to implement robotic eye also discussed in this chapter.

Complete mathematical model of the 2-axis gimbal system is derived in third chapter using the newton's second law for rotational system. In fourth chapter, a prototype face tracking algorithm is implemented on pan tilt system using Rasberri Pi and camera sensor. Chapter 5 shows the CAD model of gimbal system designed on inventor. In chapter six, PID controller is applied to the system to get the required output on simulink.

Chapter 2

Literature review

Fast eye ball tracking in the robotic eye is very important in dynamic environment where swarm of robots coordinate with each other. Human eye is the most important sensor. A human acquires more than 80 percent of information through its eyes [1]. To acquire more information in the dynamic environment about the target, fast moving stimulus tracking is required. In robotics, camera sensor is used for tracking and to locate the target fastly. Different vision techniques have been developed over the years to improve the procuring of visual information using the robotic eye.

Optical system of humans is perfect and highly developed after thousands of years of evolution. Human visual system can lock the object in the center of eye retina even when the position of head changes drastically. Due to that significance, robotic eye algorithm has been developed for the unstructured and bumpy environments. Most of researchers studied the perfect human eye tracking system and tried to implements it on the robots using the visual control, actuators and camera sensors. Researchers have developed many robotic eye tracking systems in last 20 years focusing on the human eye working. Most of the tracking systems includes the tracking and scanning algorithms based on artificial intelligence vision controller. In AI based controller, algorithms have been made and robotic eye is trained for specific motion [2]. Training the algorithm is very lengthy process because it takes lot of time to track, focus and recognize.

Most of the systems used actuators for eye ball movements with pneumatic control system [2]. But these models only work in slow environment because of coupling with head skeleton of the robot. Decoupling of the robotic eye from skeleton is very important for stimulus tracking and locating the object. In 2006, listing law was implemented on the tendon-driven robotic eye. But the decoupling ratio was very much low. Eye ball was replaced using the moveable camera by some researchers listed in [3] but still there were large amount of coupling. Most of the existing models do not respond so quickly while walking or sudden change of direction. Strong vision

control theory and good actuators are useless until the eye ball is not synchronized with control algorithm. It is not possible with high coupling ratio with skeleton head and body.
Our project goal is to track the fast moving stimulus in a dynamic environment by considering all the frictions and disturbance factors. So, Computer vision control system is needed that can point, track and can acquire more information to recognize without disturbing the mechanical design. We have to make a design that have much less coupling with head skeleton. Decoupling will allow to move fast according the vision control commands. So it will be easy to control and track the fast moving stimulus. Gimbal based system was proposed which have the ability to decouple the camera sensor from the head.

2.1 Gimbal System Literature

In this project, modeling of gimbal with camera sensor is implemented to track the fast stimulus in dynamic environment. Nowadays, most of the gimbal systems for unnamed Aerial Vehicles are available providing the decoupling along three axis yaw, pitch and roll. These systems are used in inertial navigation, seekers of missiles and stabilization of telescope. They are also used for guidance in space ship, submarines and defense industries. Extensive work has been done on the modeling and controlling of gimbal systems. There are difference disturbances in the gimbal systems. Modeling of the gimbal systems was implemented using different control techniques including disturbances in the system.

2.1.1 Modeling of the 2-axis Gimbal system

In gimbal system, there are vibrations, mass imbalances and friction which induces the disturb torque that could lead to the destabilization of the system. There are many disturbances involved in the motion of the gimbal system.

1. **Torque Disturbance:** Torque imbalance are due to the friction, gyroscope torques, actuators and gears reactions, movement of the base and also spring torques.

2. **Joint Flexibility:** Joint flexibility including bending of axis and component displacement can cause the disturbance.

Different authors used the different control techniques by ignoring some disturbances to simplify the mathematical model of the system.
In 2001, Bertil Ekstrand [4] modeled the gimbal system upon mass unbalance condition. This includes the inertia and constraint disturbances. In this it is identified that the cross coupling between the axis can be eliminated by certain inertia conditions. Maher Abdo [5] in 2013 derived the modeling equations of gimbal system with static unbalance. It has been explained that the

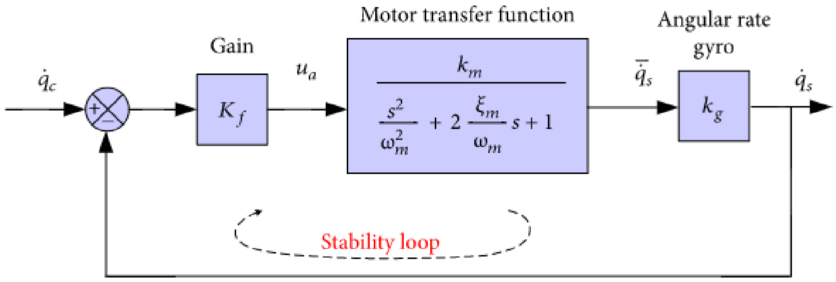

Figure 2.1: The single-axis stability loop contains a torque motor.

static unbalance occurs due to offset between the pivot center of gravity. With static unbalance, torque balance is calculated and incorporated in the control loop.

Reference [6] the effect of moving gimbal and gyroscope of magnetically suspended gimbal. In this dynamic compensation filter have been used to improve system stability. In 2018, Shixiang Liu [7], derived the modeling of the gimbal system with gyro and mass unbalance. Torque disturbance was developed on the basis of the working principle of seeker two-loop steady tracking theory. A single axis stability loop block diagram with angular rate gyro is shown in figure 2.1.

Other researchers also done the complete modeling of 2-axis gimbal system through Newton-Euler recursive method. They also ignore some unbalances to simplify the equations and to make system controllable.

2.2 Techniques

Researchers put efforts in controlling gimbal system and developed different techniques for its stabilization control design. In 1998, Li et al performed the nonlinear induced disturbance rejection in the system via Linear Quadratic Gaussian (LQG) algorithm. In linear quadratic gaussian, a quadratic cost function is minimized when the plant has random initial conditions, white noise disturbance input, and white measurement noise.

Some reserachers in [?] controlled the two axis gimbal system using fuzzy PID controller. Its equations are derived using the Lagrange equation and stabilization system is looped with torque disturbances and cross coupling. Controller loop diagram is shown in figure 2.

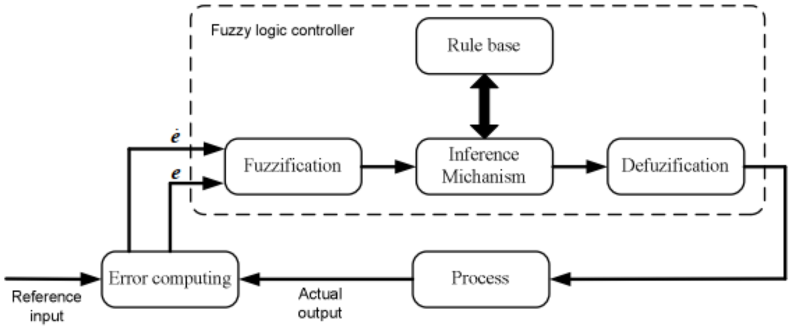

Figure 2.2: Fuzzy logic controller

In 2002, Waldmann [?] used the Extended Kalman filter (EKF) for the estimation of LOS rate from measurements of relative angular displacement between seeker gimbals and a lowcost strap down inertial unit. Robust control for inertial disturbances rejection in 2-axis gimbal system was done through PI controller and H controller designed in 2010 [?].

Reference [?] discussed the adaptive control of gimbal system with auxiliary error structure. In 2017, one axis gimbal system for gyros of different specifications was modeled with state dependent friction compensation without mass unbalance effect. In guided missile, gimbal seeker is an essential part. But it has nonlinear dynamics of seeker so it is difficult to control and tune using PID. So, sliding mode controller(SMC) is an ideal controller for this system due to its high disturbance rejection capability.

Chapter 3

Mathematical Modelling of System

The following figure shows the 2 axis gimbal system. The two gimbals are called yaw and pitch gimbal. The camera is placed in the pitch gimbal. For mathematical modelling three inertial frames are introduced. One is for Yaw and the other two are for pitch and body. x_k,y_k,z_k and x_a,y_a,$z_a$are the yaw and pitch gimbal frame axes respectively. The x_a-axis coincides with the camera axis. The center of rotation is in the frame origin, which is assumed to be the same point for the three frames. The rotation angles v_1 and v_2 are defined in the following way.

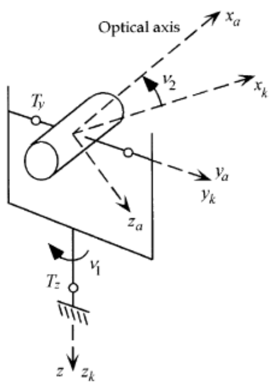

Figure 3.1: A two axis gimbal system with relative coordinates frames.

1. The body frame B is rotated to lay upon fram K by a positive angle of rotation v_1 about the z-axis.

2. The yaw gimbal frame K is moved to lay on the frame A by a positive angle of rotation v_2 about the y_k-axis.

Thus B coincides with K for $v_1 = 0$.Following transformation are associated with the rotations.

$$L_{KB} = \begin{bmatrix} cos(v_1) & sin(v_1) & 0 \\ -sin(v_1) & cos(v_1) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.1}$$

$$L_{AK} = \begin{bmatrix} cos(v_2) & 0 & -sin(v_2) \\ 0 & 1 & 0 \\ sin(v_2) & 0 & cos(v_2) \end{bmatrix} \tag{3.2}$$

L_{KB} is the transformation from B to K and L_{AK} is the transformation from A to K. We get the following angular velocities for each of the frame by using transformation.

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} ; \begin{bmatrix} p_k \\ q_k \\ r_k \end{bmatrix} ; \begin{bmatrix} p_a \\ q_a \\ r_a \end{bmatrix} \tag{3.3}$$

The inertia matrix of the gimbals are denoted by **Pitch Gimbal** is shown below

$$J_A = \begin{bmatrix} J_{ax} & D_{xy} & D_{xz} \\ D_{xy} & J_{ay} & D_{yz} \\ D_{xz} & D_{yz} & J_{az} \end{bmatrix} \tag{3.4}$$

Similarly the inertial matrix for **Yaw Gimbal** is shown below

$$J_K = \begin{bmatrix} J_{kx} & d_{xy} & d_{xz} \\ d_{xy} & J_{ky} & d_{yz} \\ d_{xz} & d_{yz} & J_{kz} \end{bmatrix} \tag{3.5}$$

The moment of inertia are denoted by J and product of inertia is denoted by D and d. center of a gimbal is supposed to be in the common center of rotation, by assuming that the gimbal has no mass unbalance.

The following torque relations are used for the gimbal system.

- T_y total external torque about the pitch gimbal y_a-axis
- T_z total external torque about the yaw gimbal z_k-axis:

3.1 Dynamic Equation of Motion

The dynamic equation of motion is obtained by the Euler angular relationship an the angular momentum of rigid body. Thus torgue is given by

$$T = \frac{dH}{dt} + w * H \tag{3.6}$$

T is the sum of external torque, **H** is the angular momentum and **w** is the angular velocity.

$$w * H = \begin{bmatrix} x & y & z \\ p & q & r \\ H_x & H_y & H_z \end{bmatrix} \tag{3.7}$$

where total angular momentum is expressed as

$$H = J_k + L_{AK}^T x J_A \tag{3.8}$$

3.2 Pitch Channel

The equation of motion for the pitch gimbal can be obtained using the law of inertia. The assumption is taken as the gimbal has no mass unbalance. Then we have

$$J_{ay} \dot{q_a} = T_y + (J_{az} - J_a ax) p_a r_a + D_{xz}(p_a^2 - r_a^2) - D_{yz}(\dot{r_a} - p_a q_a) - D_{xy}(\dot{p_a} + q_a r_a) \tag{3.9}$$

The above equation shows that certain approximation can be done by considering the symmetrical conditions.

$$D_{xy} = D_{xz} = D_{yz} = 0 \tag{3.10}$$

Assuming further that

$$J_{ax} = J_{az} \tag{3.11}$$

Equation 6 becomes

$$J_{ay} \dot{q_a} = T_y \tag{3.12}$$

Neglecting external disturbance torques, T_y represents only the motor torque. Then equation 9 is the desired equation of motion, in the sense that no disturbances influence the motion about the pitch axis.

3.2.1 Block Diagram

The pitch equation 6 can be represented by block diagram below

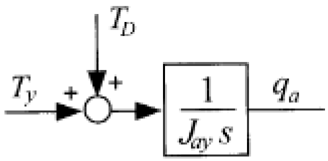

Figure 3.2: Block Diagram of Pitch Channel.

The disturbance term can be written as follow:

$$T_D = (J_{az} - J_a ax)p_a r_a + D_{xz}(p_a^2 - r_a^2) - D_{yz}(\dot{r_a} - p_a q_a) - D_{xy}(\dot{p_a} + q_a r_a) \tag{3.13}$$

Thus, for the pitch channel we have a simple model: the motor torque and the disturbance torques are inputs to an integrator which includes the moment of inertia J_{ay}, and the output is the pitch angular velocity q_a.

3.2.2 Cross Coupling

As the pitch gimbal coordinates are related to yaw by the following transformation. We can write

$$p_a = p_k cos(v_2) - r_k sin(v_2) \tag{3.14}$$

$$q_a = q_k + \dot{v_2} \tag{3.15}$$

$$r_a = p_k sin(v_2) + r_k cos(v2) \tag{3.16}$$

By using the above relations the disturbance term can be written as :

$$T_D = T_B + T_C \tag{3.17}$$

where

$$\begin{aligned} T_B = &-(D_{yz} sin(v_2) + D_{xy} cos(v_2))(\dot{p_k} + q_k r_k) + (D_{yz} cos(v_2) - D_{xy} sin(v_2))p_k r_k \\ &+ [(J_{az} - J_{ax}) cos(2v_2) - 2D_{xz} sin(2v_2)]p_k r_k + \frac{1}{2}[(J_{az} - J_{ax}) sin(2v_2) + 2D_{xz} cos(2v_2)]p_k^2 \end{aligned} \tag{3.18}$$

and

$$T_C = (D_{xy}sin(v_2) - D_{yz}cos(v_2))\dot{r_k} - \frac{1}{2}[(J_{az} - J_{ax})sin(2v_2) + 2D_{xz}cos(2v_2)]r_k^2 \tag{3.19}$$

From this separation of T_D into two parts, it is clear that the yaw gimbal may influence the pitch gimbal irrespective of body motions.

3.3 Yaw Channel

The following equation of motion can be derived for yaw gimbal.

$$J_k\dot{r_k} = T_z + T_{d1} + T_{d2} + T_{d3} \tag{3.20}$$

where

$$J_k = J_{kz} + J_{ax}sin^2(v_2) + J_{az}cos^2(v_2) - D_{xz}sin(2v_2) \tag{3.21}$$

$$T_{d1} = [J_{kx} + J_{ax}cos^2(v2) + J_{az}sin^2(v_2) + D_{xz}sin(2v_2) - (J_{ky} + J_{ay})]p_kq_k \tag{3.22}$$

and

$$\begin{aligned} T_{d2} = -[d_{xz} + (J_{az} - J_{ax})sin(v_2)cos(v_2) + D_{xz}cos(2v_2)](\dot{p_k} - q_kr_k) \\ - (d_{yz} + D_{yz}cos(v_2) - D_{xy}sin(v_2)) \end{aligned} \tag{3.23}$$

$$\begin{aligned} T_{d3} = \ddot{v_2}(D_{xy}sin(v_2) - D_{yz}cos(v_2)) + \dot{v_2}[(J_{ax} - J_{az})(p_kcos(2v_2) - r_ksin(2v_2)) \\ + 2D_{xz}(p_ksin(2v_2) + r_kcos(2v_2)) + (D_{yz}sin(v_2)) + D_{xy}cos(v_2)(q_a + q_k) - J_{ay}p_k] \end{aligned} \tag{3.24}$$

The approximation of symmetry and other static equilibrium assumption can be used to make the modelling simplify.

3.3.1 Block Diagram

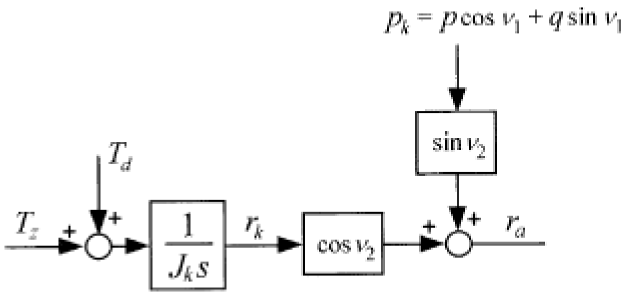

Figure 3.3: Block Diagram of Yaw Channel.

3.4 Linearization

As it can be seen from the mathematical modeling that the system is nonlinear and cross coupled. Thus it can be linearized by using the techniques available for linearization like Jacobian and Taylor series. The linearization is carried out at a singular point which in our case is of 4 values (0,0,0,0). As we are dealing with two nonlinear second order differential equations we can spilt them into 4 linear first order differential equations. This is done by using Jacobian Method which is described below.

We can get the A matrix and B matrix of state space system through this Jacobian matrix.

$$A = \begin{bmatrix} \frac{df_1}{dx_1} & \frac{df_1}{dx_2} & \frac{df_1}{dx_3} & \frac{df_1}{dx_4} \\ \frac{df_2}{dx_1} & \frac{df_2}{dx_2} & \frac{df_2}{dx_3} & \frac{df_2}{dx_4} \\ \frac{df_3}{dx_1} & \frac{df_3}{dx_2} & \frac{df_3}{dx_3} & \frac{df_3}{dx_4} \\ \frac{df_4}{dx_1} & \frac{df_4}{dx_2} & \frac{df_4}{dx_3} & \frac{df_4}{dx_4} \end{bmatrix} \tag{3.25}$$

Similarly,

$$B = \begin{bmatrix} \frac{df_1}{du_1} & \frac{df_1}{du_2} \\ \frac{df_2}{du_1} & \frac{df_2}{du_2} \\ \frac{df_3}{du_1} & \frac{df_3}{du_2} \\ \frac{df_4}{du_1} & \frac{df_4}{du_2} \end{bmatrix} \tag{3.26}$$

State Space model is:

$$\dot{x}(t) = Ax(t) + Bu(t) \tag{3.27}$$

$$y(t) = Cx(t) + Du(t) \tag{3.28}$$

Here D matrix is 0 and as we want x1(Yaw angle) and x3(pitch angle) at the output so C matrix will be:

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{3.29}$$

In the next chapter, yaw and pitch torque equations are used to get the final state space matrices. Then state space is modeled is simulated using simulink on Matlab.

Chapter 4

Software Implementation and Controller Design

The modeling equations derived in the last section is simulated in the matlab and we applied different controllers to control our response.

Mathematical equations of yaw and pitch axis is loaded in the matlab. State space calculation algorithm is also implemented in the Matlab. From the values of different constant inertia and disturbances terms, final matrices of state space equations are calculated which represent the output of our gimbal.

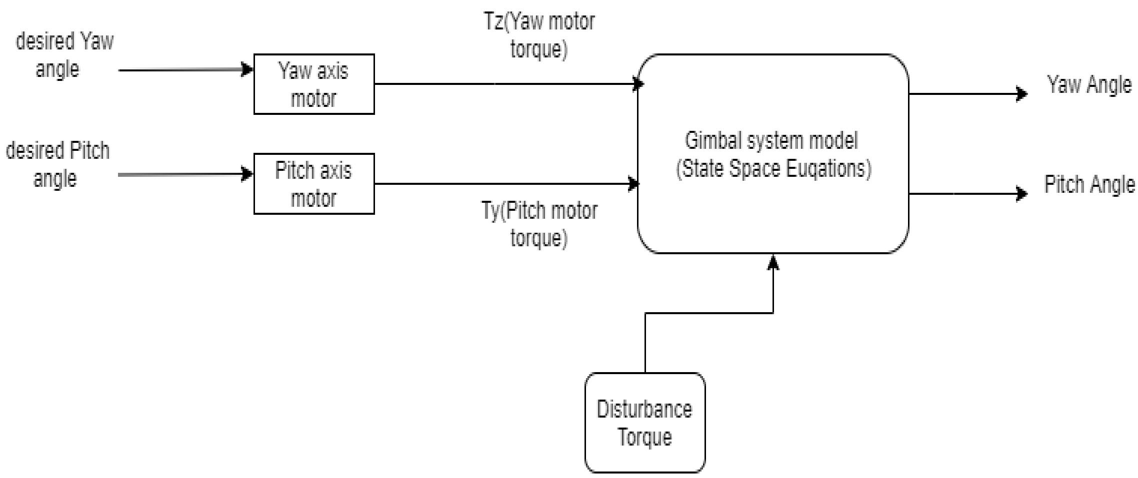

Figure 4.1: Block diagram

Gimbal model block diagram is shown in figure 4.1. Desired pitch and yaw angle is provided to the motor with in form of step voltage. Then motors at each axis (yaw and pitch) will provide torque to the system and accordingly our axis will reach to the desired angle.

4.1 State space equations

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1300 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -2700 & 0 \end{bmatrix} \tag{4.1}$$

$$B = \begin{bmatrix} 0 & 0 \\ 1 & -2 \\ 0 & 0 \\ -2 & 4.33 \end{bmatrix} \tag{4.2}$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{4.3}$$

$$D = 0 \tag{4.4}$$

We convert this system into the Simulink blocks. The detailed implementation of the system is shown in below figure.

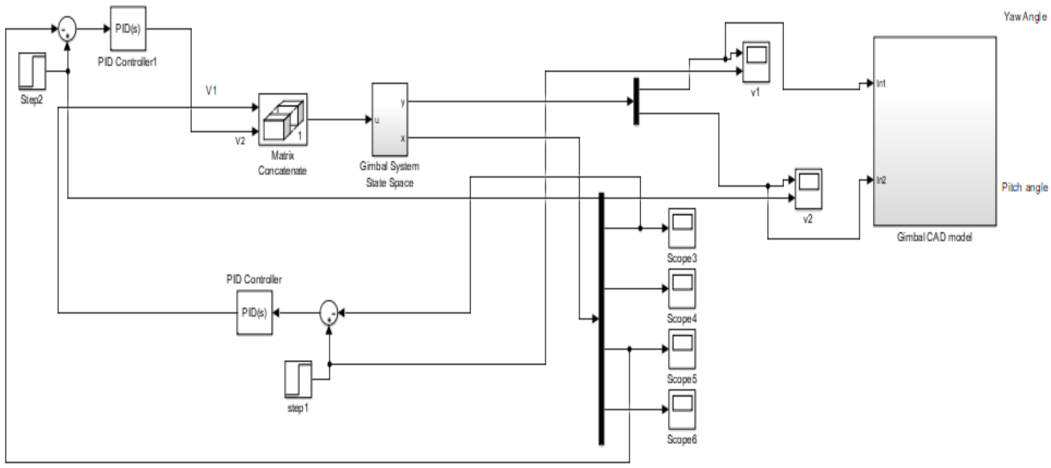

Figure 4.2: Simulation diagram of overall system

4.1.1 Gimbal System State Space Block

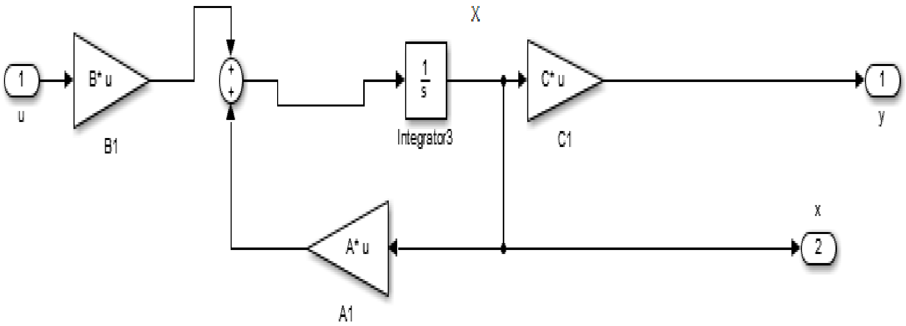

Figure 4.3: State Space implementation

4.1.2 Simulink Model of Mechanical Gimbal CAD model

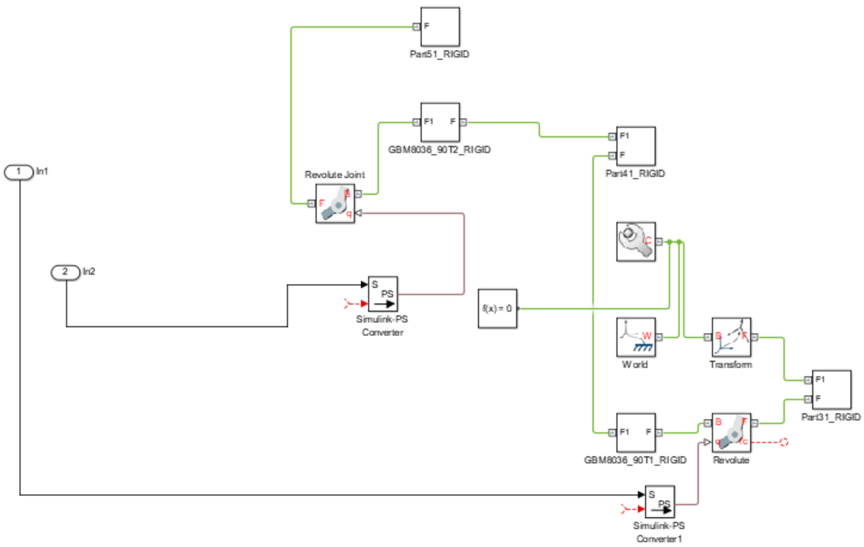

Figure 4.4: Simulink Model of mechanical gimbal

All the outputs of the state space system are displayed on the scope and applied in the control loop with PID controller

4.2 Control Objective

Our control objective is that our output of the system should follow the desired input. As from figure 4.2, we have yaw and pitch angle at the output and our required yaw and pitch angle at the input. Our goal is that output angles should follow our given input angles. For this we used the PID controller as shown in figure 4.2.

4.3 PID Controller

Proportional integral derivative (PID) controller is a basic controller configuration which derives the output to the desired level or value. It is used in most of industrial processes because it is very effective and understandable. We used the proportional, derivative and integral of the signal to generate the control signal.

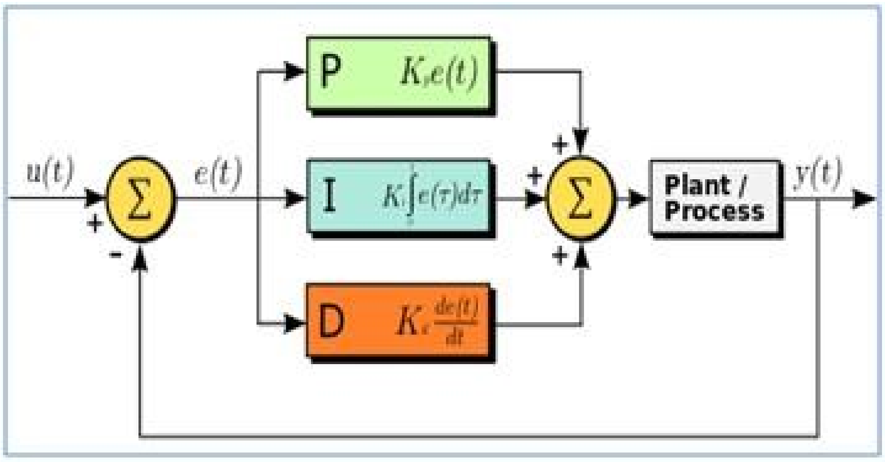

Figure 4.5: Working of PID controller [?]

In above figure, e(t) is the error signal. u(t) is the desired input and y(t) is the actual output. Error signal is given to the PID and its output is given to the plant. The new output y(t) is generated which is fed back to the comparator. In this way, output followed our desired input signal.
Transfer function of PID controller is:

$$K_p + \frac{K_I}{s} + K_d s = \frac{K_d s^2 + K_p s + K_t}{s} \tag{4.5}$$

Kp is the proportional gain to the error(u(t)-y(t)). When error is large, control output signal will be large with gain Kp. When there is no steady state error then it will be zero and stop working. Ki is the integral gain and it accounts for past values of error. It sums the past values of error so even small error causes the integral term to increase slowly and try to make the error zero as soon as possible.
Kd represents the derivative gain which predict the future values of the error. It reduces the error by generating control signal by taking the rate of change error. When we increase the derivative time, control signal will react more powerfully, so response of the system increases.
We implemented the PID controller from the Simulink block in our gimbal system to get our desired output.

4.3.1 Tuning of PID controller

In Simulink PID block, we have auto tuning mechanism. Auto-tuned mechanism is used to tune the system. To get more better response, values of Kp, Ki and Kd are adjusted on hit and trial basis. Step signal is given to the input and the output are displayed on the scope in Simulink.

4.4 Working

Detailed diagram of Simulink implementation of our system is shown in figure 4.2. In this Gimbal system state space block represents the final state space equations with all the disturbances. As we have two inputs two outputs (TITO) system, we have to design two PID controller for each output. So we have made the two controller loops as shown in figure 6.2.
In figure 6.2, v1 is the yaw angle and v2 is the pitch angle. Step1 input is the reference input for the yaw angle and the step2 is the reference input for the pitch angle. Tunned PID controllers are used to minimize the error between reference input and output. We gave the reference inputs through step1 and step2 and output is taken from the y terminal of the gimbal system state space block. These outputs yaw and pitch angles are given to the respective inputs of the axis in the cad model of gimbal. So we can anlayze our output response with the movement of gimbal.

4.5 Results and Discussion

Results of the outputs were taken for different inputs. For fast stimulus detection step input is given.

1. Step Input

The output waveform of the angles with step input is shown in below figures.

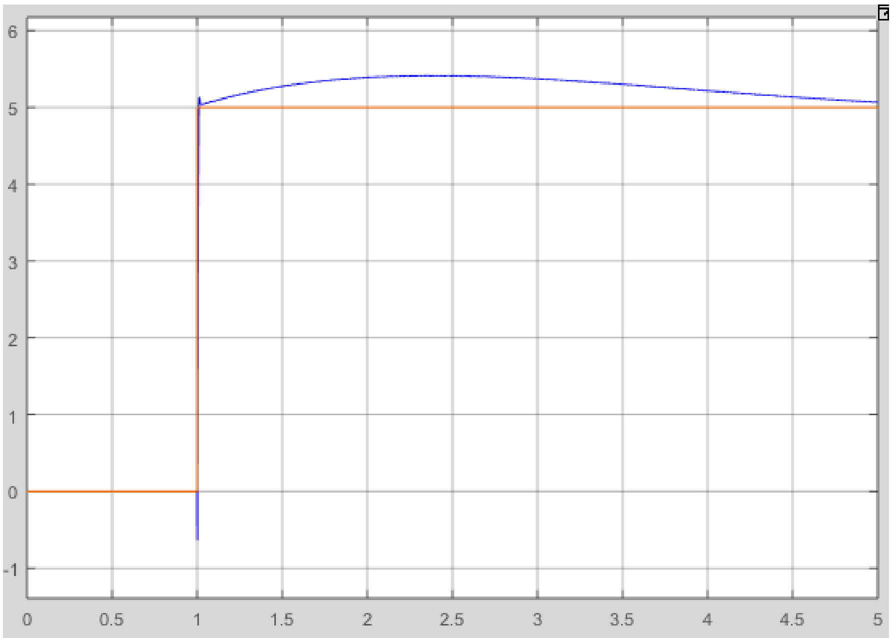

Figure 4.6: Yaw angle

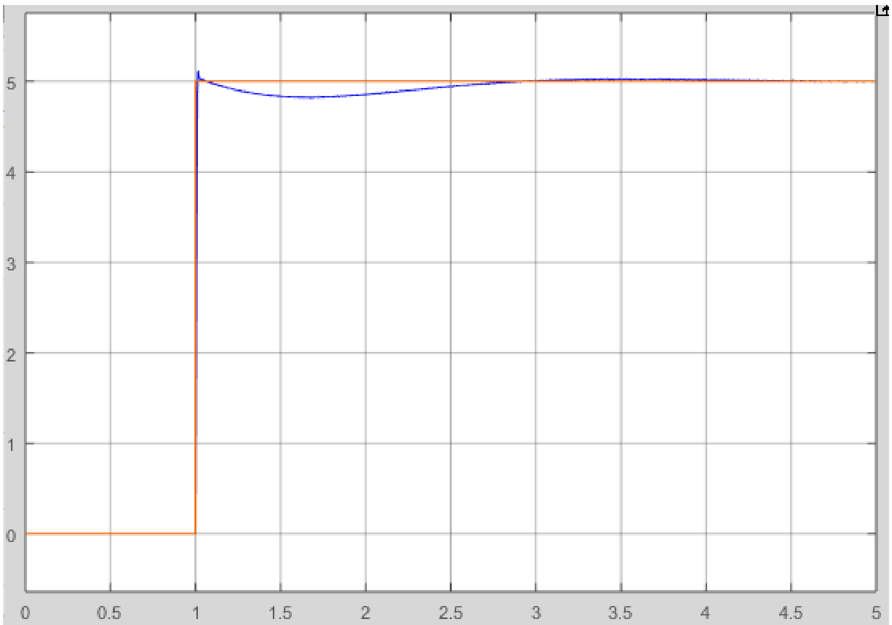

Figure 4.7: Pitch angle

From the above figure, we can see that the output signal which is the angle of yaw/pitch axis follows our desired step signal. We can see that rising time is small but there are some overshoots but eventually we achieved our desired output.

In yaw angle, we have large settling time and in case of pitch axis we have small settling time.

2. Ramp Input

Now we have given ramp input for 4 seconds then it become constant. Output waveforms for each channel is shown in figure below.

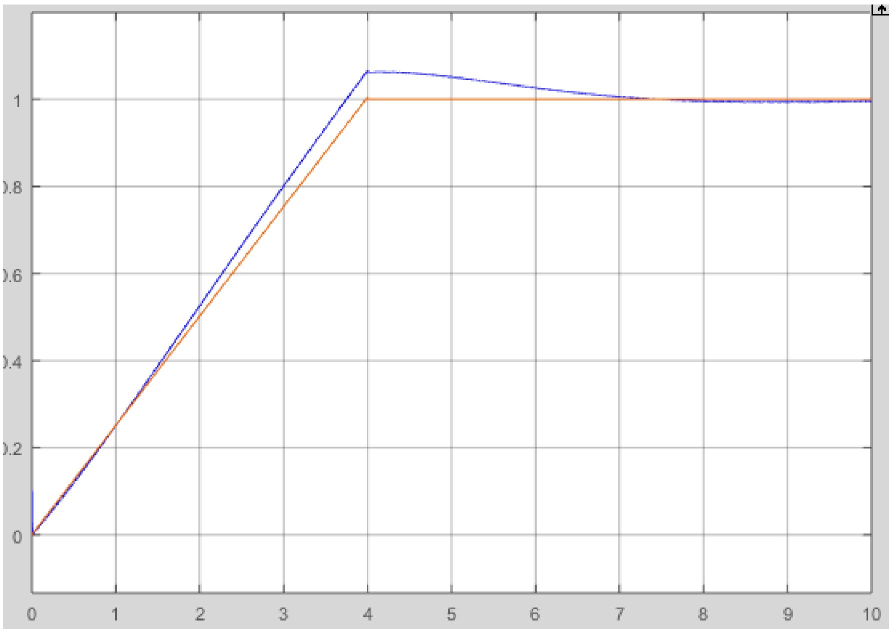

Figure 4.8: Yaw angle

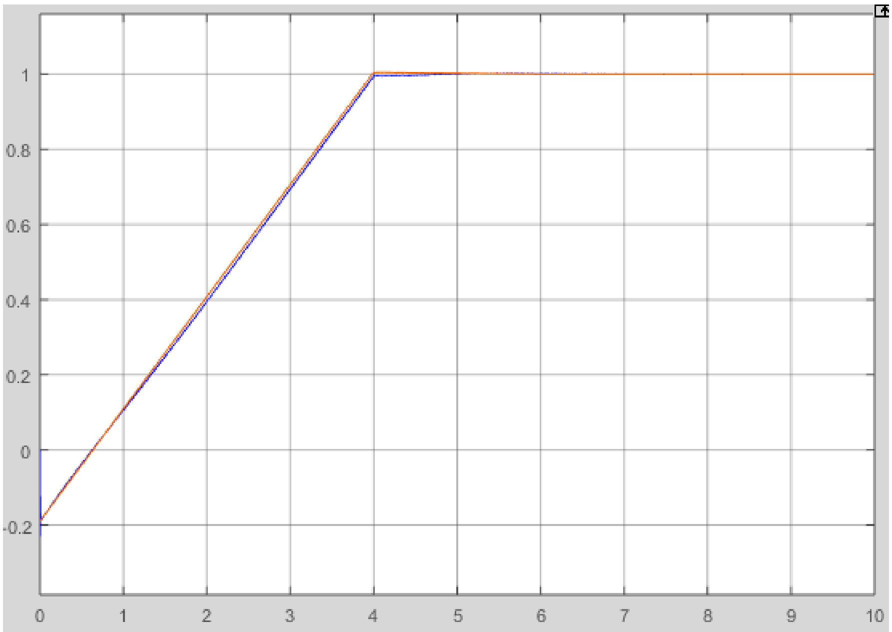

Figure 4.9: Pitch angle

Each output of the gimbal system follows the reference input. We have some overshoot as in yaw channel output but eventually steady state error becomes zero. So stimulus detection is achieved.

So from these output it is clear that our system is capable of tracking the fast moving stimulus. As disturbance of environment is very much less on the gimbal system, so we can recognize and track our object using the system and camera sensor as eye ball.
Now we implemented the some of tracking algorithms using python and raspberry Pi on prototype pan tilt system which is discussed in the next chapter.

Chapter 5

Experimental Implementation

5.1 Mechanical Design

First we made the CAD model of our gimbal system to simulate it on the Matlab. The final design is shown in the figure below.

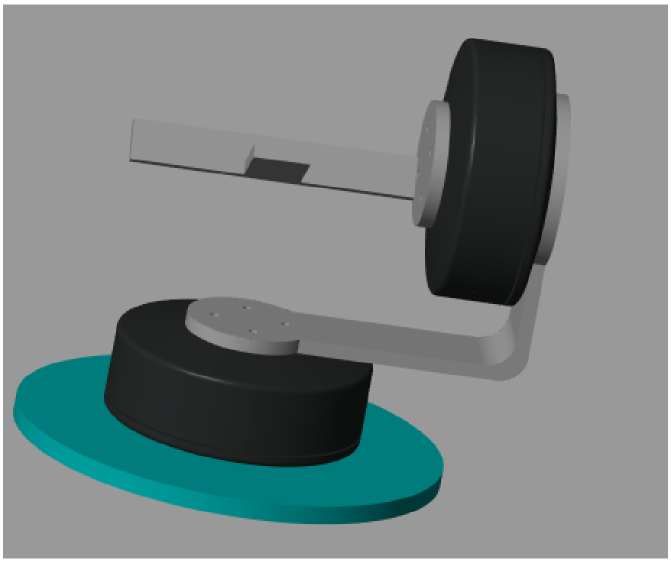

Figure 5.1: CAD model of mechanical gimbal

In this model two axis of gimbal(yaw and pitch) is shown and base is fixed. We have brushless DC motor for each axis. Then we import our gimbal CAD model into the Simulink. Simulink automatically generated its equivalent simulink block model which can be controlled. So we analyzed and understood its block model to give input to the yaw and pitch axis. So we linked this model with our state space model of gimbal and observed the motion of yaw and pitch axis. Gimbal was tracking the point according to the reference input value.

5.2 Raspberry Pi Implementation

A simple prototype of 2 axis gimbal system is implemented on FPV Pan tilt gimbal on Raspberry pi. This simple project demonstrate the basic level techniques of Computer Vision. This assembly is available online on Aliexpress and bought from it via online delivery for the price of 7 dollars.

5.2.1 FPV Pan tilt Gimbal

This is a prototype assembly of pan tilt gimbal which allows two axis movement along yaw and pitch axis.

Figure 5.2: FPV Pan tilt Assembly.

The figure below show the two axis movement of FPV gimbal. The Raspberry pi camera is mounted on the inner gimbal which is a pitch gimbal.

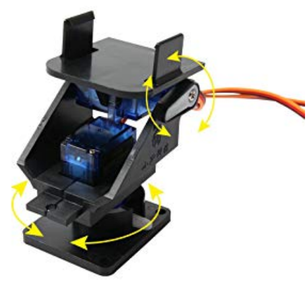

Figure 5.3: FPV Pan tilt Assembly Rotation.

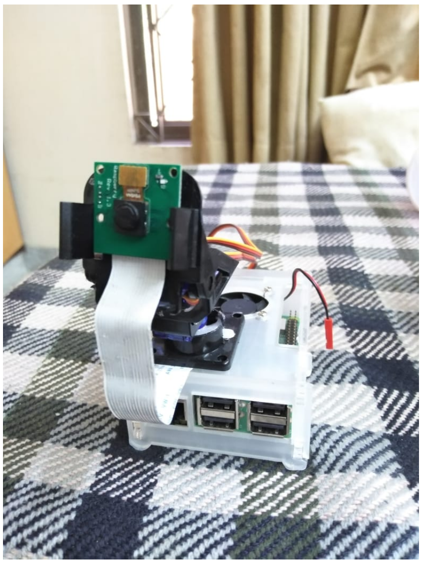

Figure 5.4: Raspberry PI Assembly of FPV Gimbal.

The camera used in this case is of 5MP and connected to the raspberry pi via bus for communication. The whole assembly is bought from China via Aliexpress for 80 dollars.

5.2.2 Raspberry Pi

We use Raspberry Pi to implement the algorithm which was developed by using basic level techniques. A Raspberry Pi is a single board computer and has an excellent processing power. It has the ability to support various computer vision and image processing algorithms. We use Raspberry Pi 3B+ model. This board has 1.4GHz 64-bit quad-core processor. The board has the ability to connect a large number of the peripherals and devices like camera and sound systems etc. The board has a 40 pin GPIO which give it a power to attach additional circuitry.

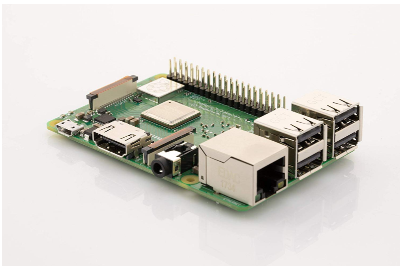

Figure 5.5: Raspberry pi SOC board.

5.2.3 Raspberry Pi Camera

Below is the figure of Raspberry camera module which is of 5MP and high resoultion of 1080p. It has 5 megapixel OV5647 sensor in an adjustable-focus module.

Figure 5.6: Raspberry pi SOC board.

5.2.4 Face Tracking With FPV gimbal

The FPV gimbal has two specific movements. One is in the Yaw direction and the other one is in the pitch direction. Thus it can be used to track a certain object in its view. A camera is mounted on the Raspberry Pi assembly. The gimbal servo motors are connected to the GPIO of the Raspberry Pi. The Raspberry Pi is programmed to get the data from the camera module.

5.2.5 Algorithm

The algorithm is quite simple which is to first detect the object and then pan and tilt to make the object remain in the center of the frame. This involve image processing. The desired feature must be extracted from the image and then processed to apply the desired command to the motors. The image proecessing is done on the Raspberry Pi.

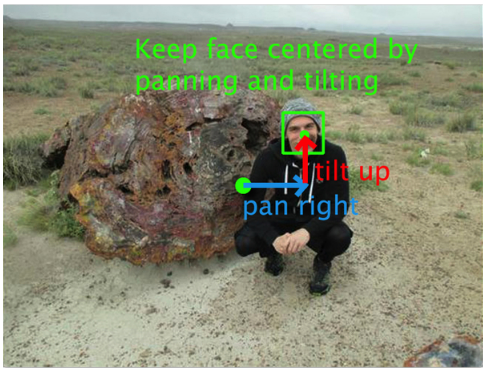

Figure 5.7: Pan and tilt technique.

5.2.6 Face Detection

The face detection is done using an image classifier which in this case is Haar Cascade Algorithm. The Haar Cascade algorithm is a pretrained classifier for face tracking. It can be deployed to any of the operating system for detections and tracking of faces. The classifier can be trained for a specific person face.

5.2.7 Circuit Schematics of Assembly

Below is the circuit assembly of Gimbal system. The two servo motors are connected to the GPIO pins of the Raspberry PI. The pins are mentioned in the code.

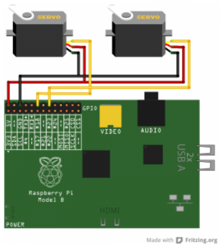

Figure 5.8: Circuit Schematic of Raspberry pi.

5.2.8 Explanation

The first code is used to train the gimbal for a particular persons face. The second code is used for real time tracking of the face by panning and tilting. The code is the implementation of the algorithm mentioned above.

Chapter 6

Conclusion and Future Work

In order to design a robotic eye for fast moving stimulus, we first developed a theoretical understanding then mathematically modeled it. The nonlinear system is then linearized by jacobian after which a controller is designed and successfully simulated. The next step is to implement the system practically. For this purpose we have done the research and designed a strategy for implementation. For this process we first have to analyzed the system requirement. The ultimate requirement for our project are as follow:

- Our system must be robust so that it can track the target precisely and timely.
- For object tracking, the image quality must be very high, thus camera sensor must be of high quality.
- A high processing power is required by the algorithm so we need an embedded system which must be capable of doing so.
- The cost must not exceed the budget allotted for the project.

Keeping the above requirements in mind we have done the research. And came to the following conclusions

- For robustness the actuators must give a quick response and must be efficient.
- Camera should able to detect the fast moving stimulus and thus have a high resolution and focus.
- The image processing requires a high processing power for our project as it has to implement AI algorithm .

Upon analysis we choose the following equipment that we will be used for implementation.

Figure 6.1: The fisheye camera for the system.

Figure 6.2: Brushless gimbal motors for low speed precise movements.

The above two resources are available in the market for a price 100 dollars. The specification of these equipments are as per our project requirement.

For processing we will use laptop for image processing and arduino for gimbal controller.

6.1 Conclusion

This thesis report is the summary of our work on robotic eye that we learned and implemented up till now. We have applied what we have learned up till now about the gimbal and implemented by doing simulations. We have used a practical value for the sake of simulation of actuators and sensors. The results are quite satisfactory as shown in the simulation section. We are ready to implement practically and ready to apply the techniques that we have learned so far in our research.

Bibliography

[1] G. Cannata and E. Grosso, “On perceptual advantages of active robot vision,” *Journal of Robotic Systems*, vol. 16, no. 3, pp. 163–183, 1999.

[2] K. Nickel and R. Stiefelhagen, “Visual recognition of pointing gestures for human–robot interaction,” *Image and vision computing*, vol. 25, no. 12, pp. 1875–1884, 2007.

[3] C. Gielen, S. Gabel, and J. Duysens, “Retinal slip during active head motion and stimulus motion,” *Experimental brain research*, vol. 155, no. 2, pp. 211–219, 2004.

[4] B. Ekstrand, “Equations of motion for a two-axes gimbal system,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 37, no. 3, pp. 1083–1091, 2001.

[5] M. Abdo, A. Toloei, A. R. Vali, and M. R. Arvan, “Cascade control system for two axes gimbal system with mass unbalance,” *International Journal of Scientific & Engineering Research*, vol. 4, no. 9, pp. 903–913, 2013.

[6] X. Chen and Y. Ren, “Modal decoupling control for a double gimbal magnetically suspended control moment gyroscope based on modal controller and feedback linearization method,” *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, vol. 228, no. 13, pp. 2303–2313, 2014.

[7] S. Liu, T. Lu, T. Shang, and Q. Xia, “Dynamic modeling and coupling characteristic analysis of two-axis rate gyro seeker,” *International Journal of Aerospace Engineering*, vol. 2018, 2018.