

A Neuro-Fuzzy Based Approach to Object Tracking and Motion Prediction

Engr. Simon Samuel
Department of Electrical and
Electronics Engineering
Modibbo Adama University of
Technology Yola, Adamawa
State, Nigeria

Engr. Ibrahim A. Usman
Department of Electrical and
Electronics Engineering
Modibbo Adama University of
Technology Yola, Adamawa
State, Nigeria

Engr. Baams Baamani Alfred
Department of Electrical and
Electronics Engineering
Modibbo Adama University of
Technology Yola, Adamawa
State, Nigeria

Abstract: In this Paper, object tracking system model was developed using Neuro-Fuzzy hybrid based approach to predict the trajectory of an object moving around a scene. Servo motors were used to perform high-precision positioning in azimuth and elevation directions, fuzzy logic is applied to control the position servo motors via feedback. A Neuro-Fuzzy hybrid approach is used to design the fuzzy rule base of the intelligent system for control. In particular, ANFIS methodology was used to build a Sugeno fuzzy model for controlling the servo motor position carrying charge couple device camera (CCD) on a chaotic trajectory. An advanced test bed is used in order to evaluate the tracking properties and the robustness of the ANFIS controller operations. However, the variations of the Mechanical configuration of the drive, which is common to these two applications, can lead to error in object positioning before segmentation. The result for the azimuth and elevation time responses show that the rise time t_r reduces to 0.1 and 0.3, respectively. The settling time decreases to 0.5 for the motors with ANFIS controller, the delay time reduces to 0.1 for both motors. Steady state was reached. Conclusively, ANFIS controller output was the best in terms of faster rise time, settling time, reduced delay time and object position stabilization.

Keywords: ANFIS, Charge couple device (CCD)-Positioning System, DC servomotor, Segmentation.

1. Introduction.

Object tracking and motion prediction are ways of estimating moving object path and position as it moves/stops in an environment. It plays a key role in an intelligent visioning for environmental monitoring during surveillance especially in military, traffic monitoring and sport video analysis. In tracking, a system approaches a moving object (also called target) matching its location and velocity. Tracking system in literature consist of three parts; a controller that receives information on the position of object through a sensor, a dummy moving mechanism which follows an object based on controller directive, and a system vision which captures the object picture as it moves around a scene e.g a camera[1]. A major problem to object tracking is the uncertainty associated with the environments within which the system have to operate due to illumination, changing appearance patterns of the object and the scene, abrupt object motion, non-rigid object structures, object-to-object and object-to-scene occlusions, and camera motion stabilization. To address the fore mentioned challenges the controller must denote the ability to learn about the operating environment, the moving mechanism directed by the controller must be prompt or without delay to follow the object, finally, the system vision which in most case the camera must be clear to present the target object when segmented through coloured image processing [2].

The study is aimed at tracking object operating on a sequence of video images provided by camera through the application of Neuro-fuzzy logic system.

The objectives of this work are; to use Neuro-fuzzy techniques to develop a rule based fuzzy system as an efficient and robust approach for object tracking on a sequence of video image and predict position of object on a trajectory.

2. Related Works

Object tracking was defined as approaching a moving object (also called target) matching its location and velocity. The tasks have a wide range of applications. We shall review some of the methods outlining clearly the limitations that necessitate the hybrid method used. The early tracking methodologies uses background subtractions; in these methods, the authors utilize and optimized threshold systems to obtain the behavior of the moving object, centroid of the object was computed to analyze the position of the moving object. However, in this work, the cameras' used were fixed and cannot follow moving object [5]. Another set of authors used neural network to track an object, the method utilize one to one organizing map to model a dynamic background and tracks object from video as well as static object/image. It was noticed that, this method involves computational complexity and this makes tracking difficult [6]. In a separate work, a particle filter approach was performed, the work established to track and predict non-linear, non Gaussian moving object in a certain environment. In this work, histogram colour of every object was obtained and sampled. The particle approach was found to be limited by the volume of combined foreground predicted information. Another author used Kernel based algorithm, in this method, tracking was performed by calculating moving object velocity and shape represented by a potential object region, from one frame to the next frame. Although, the method have a good directional selectivity it was observed that it has limitation since it used real filter which cannot track non-linear motions [8]. In another review the author [9], worked on baseline approach, the approach received input and image that depicts object of interest O and a video V of N frames and predict instances of this object. The similarities against all the frames of the video are evaluated stored in bounding box the Coordinates of the upper-left corners (x, y) calculated to tracks object. The major limitation observed in the method is that the process is slow.

The remaining parts of the work are organized as follows, 2 system description, 3 modeling and simulation of servomotor and ANFIS controller, 4 result and discussions, 5 summary conclusion and recommendation for further studies.

3 System Description

Fig.1 is the complete block diagram for object tracking system. The tracking system is made up of one charge couple device (CCD) camera, two servomotors and an ANFIS controller. The CCD video camera is mounted on elevator motor shaft to pan the camera field of view, whereas, the azimuth motor moves both the elevator motor with the camera to tilt the camera field of view. The video output of CCD camera and the motion/position of camera are the feedback signal to the input and output of the tracking system respectively. Consequentially, CCD camera is used as an image sensor and to measure the position of moving target θ_r . The first input to the summer is the set position θ_d , i.e the desired position of the azimuth or elevation motor. The second input is the feedback signal, the current position θ_r of

the object from the camera. The difference between the two inputs θ_r and θ_d , is called position error signal $\theta(e)$, is given to the controller that reads the signal and produce appropriate output signal ω . The controller output reaches the motor driver, which produces a proportional output to rotate the respective motor in either direction according to the sign of the error signal. As the desired position of the object is approached, the error signal is reduced to zero and motor stops. In order to have stable focus with the CCD camera, the targets are automatically locked whenever the camera sensor senses that the object is at the centre of the camera field of view. Target locking is the act of maintaining the target in the camera sensor's center field of view. Under target motion, the target is tracked continuously and to have the object in center of image, the camera is moved in Pan (elevation) and tilt (azimuth) direction according to target position in image. The two servomotors rotate from 0 to $\pm 90^\circ$ along Azimuth and Elevation angles. Object Segmentation is carried out once the object is within the camera field of view.

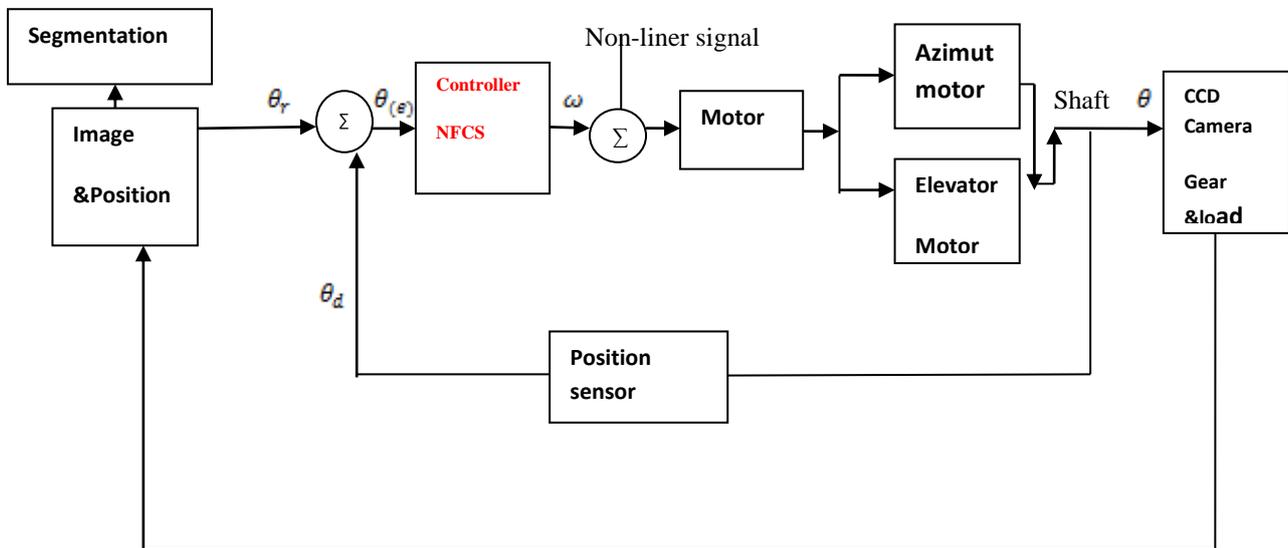


Fig.1 block diagram showing connected systems for object tracking

3.1 Modeling of D.C servomotor

First DC servomotor which positions the system vision (camera) is modeled mathematically for analytical applications. Figure 2 shows the complete parts of the servomotor.

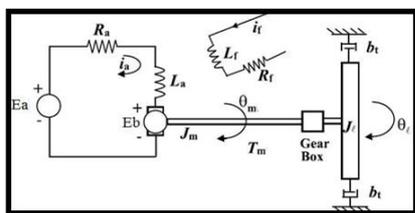


Fig 4 Armature controlled DC servo motor

For the armature controlled DC servomotor in Figure 2 the field current i_f is constant and the torque T_m generated by the DC motor shaft is given by:

$T_m = K_t i_a$ Where K_t is the given motor torque constant in (N-m/A) and i_a is the armature current in (A) [12]. The back

emf induced is directly proportional to the armature angular velocity

$\omega_m(t) = \frac{d\theta_m}{dt}$ where $\theta_m(t)$ is the angular velocity of the motor shaft thus, E_b

Where K_b is a given motor constant (Volt-sec/rad). Fig. 2

The angular speed of armature control motor $\omega_m(t)$ is controlled by armature voltage E_a . The differential equation relating armature current i_a back emf E_b is obtained by Kirchhoff's Voltage law as;

$$L_a \frac{di_a}{dt} + R_a i_a + E_b = E_a$$

since the DC servomotor shaft is connected to a gear box of given gear ratio K_g and the load which is the camera is attached to the output shaft of the gear-box. The gear ratio K_g is given by

$K_g = \frac{n_l}{n_m}$ where n_l and n_m are number of teeth on the load-side and the motor gears, respectively. The gear ratio

K_g relates the motor shaft angular position θ_m to the gearbox output shaft angular position θ_l by $K_g = \frac{\theta_m}{\theta_l}$. In addition J_L load inertia acting at the output shaft of the gearbox when reflected at the motor shaft is give by $\frac{1}{K_g^2} J_L$

Using Newton's moment balance equation at the motor output shaft [13].

$$T_m = J_m \frac{d^2 \theta_m}{dt^2} + \frac{1}{K_g^2} J_L \frac{d^2 \theta_m}{dt^2} + \frac{1}{K_g^2} b_t$$

This can be rewritten as

$$J_{\epsilon q} \frac{d^2 \theta_l}{dt^2} + b_t \frac{d \theta_m}{dt} = K_g T_m$$

Where $J_{\epsilon q} = K_g^2 J_m + J_L$ is the total load inertia reflected at the motor shaft and b_t is the rotational viscous frictional constant.

Taking the lap lace transform of the last four equations to eliminate the variables T_m, E_b , and i_a .

$$\frac{\theta_l(s)}{E_a(s)} = \frac{K_g K_t}{S(L_a J_{\epsilon q} S^2 + (L_a b_t R_a J_{\epsilon q}) S + R_a b_t + K_g^2 K_t K_b)}$$

The transfer function from input E_a to output ω_l is give by

$$\frac{\omega_l(s)}{E_a(s)} = \frac{K_g K_t}{(L_a J_{\epsilon q}) S^2 + (L_a b_t R_a J_{\epsilon q}) S + R_a b_t + \frac{\omega_l(s)}{E_a(s)} K_g^2 K_t K_b}$$

Now, assuming two real, simple roots of the characteristic equation can be obtained as P_e and P_m , using partial fraction expansion

$$\frac{\omega_l(s)}{E_a(s)} = \frac{A_\epsilon}{S + P_\epsilon} + \frac{A_m}{S + P_m}$$

Using the inverse Laplace transforms the forced response of the system (with zero initial condition) to the input $E_a(t)$ is given by;

$$\omega_l(t) =$$

$$\int_0^t (A_\epsilon e^{-p_\epsilon(t-q)} + A_m e^{-p_m(t-q)}) E_a(q) dq$$

In most practical application of armature control DC motor, $P_\epsilon \gg P_m$; that is the electrical subsystem respond considerably is faster than the mechanical subsystem. Hence, the first exponential term decay rapidly thus the respond $\omega_l(t)$ is dominated by mechanical subsystem $\frac{A_m}{S + P_m}$. in D.C servomotor the influence of electrical subsystem component $\frac{A_\epsilon}{S + P_\epsilon}$ on the response $\omega_l(t)$ is commonly neglected, viewed

as neglecting the armature inductance effect L_a , this simplification yields a first-order transfer function model which relates the DC motor load angular velocity response ω_l to the armature voltage input E_a given by

$$\frac{\omega_l(s)}{E_a(s)} = \frac{K_g K_t}{R_a J_{\epsilon q} S + R_a b_t + K_g^2 K_t K_b}$$

In S.I unit K_t and K_b numerical values are identical Hence, the transfer function model can be written as

$$\frac{\omega_l(s)}{E_a(s)} = \frac{K}{\tau S + 1}$$

Where K and τ are the D.C gain and mechanical time constant of D.C servomotor respectively [14].

3.1.1 Servomotor position control with position feedback

$\frac{\theta_l(s)}{E_a(s)}$ Is given as;

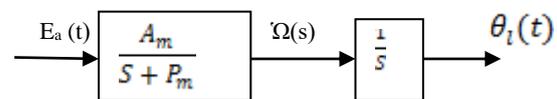


Fig 3 Servomotor transfer function model

The s-domain unit step response is

$$\theta_l(s) = \frac{A_m}{S(S + P_m)} \frac{1}{s}$$

The final value of the response is $\lim_{t \rightarrow \infty} \theta_o(t) = \lim_{s \rightarrow 0} s \theta_l(s) = \infty$. That is, the position response is unbounded. In order to control the output position to follow an input command, consider the addition of a position feedback and feedback is given by;

$$E_a(s) = K_p [\theta_i(s) - \theta_l(s)] \cdot \Omega_c(s)$$

As shown in Figure 4 the purpose of this system is to have the output angle $\theta_l(t)$ follow the input angle $\theta_i(t)$ [15].

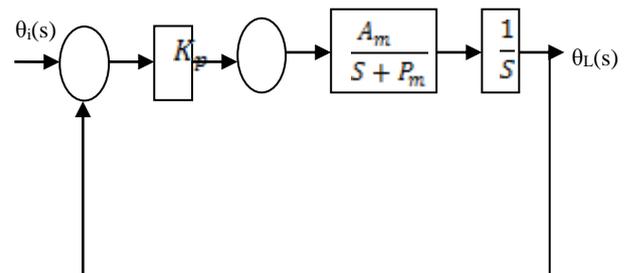


Fig.4 Servomotor position control using position feedback

Applying Mason's gain formula and show that the overall transfer function is

$$\frac{\theta_L(s)}{\theta_i(s)} = \frac{K_p A_m}{s^2 + P_m s + K_p A_m}$$

Using parameters obtained from servo motor vendor's in table 1, A_m and P_m are calculated for both azimuth and elevation servo motor.

Table 1
Parameters of Model with D.C Servomotors

PARAMETER	DEFINATION	ELEVATION	AZIMUTH
Δ	Motor Dampening constant [Nms/rad]	0.04	0.01
J_m	Motor Inertial Constant [Kgm ²]	0.00001	0.001
J_l	Load inertia [Kgm ²]	0.0044	0.00001
$J_{eq} = K_g^2 (J_m + J_{tac}) + J_l$	Equivalent inertia [Kgm ²]	0.00012	0.000013
K_B	Back emf Constant [Vs/rad]	0.055	0.5
B_{eq}	Equivalent friction required to the secondary gear [Nm/(rad/s)]	0.001	0.01
K_g	Gear Ratio	0.1	0.1
K_T	Motor Torque Constant [Nm/A]	0.00006	0.5
L_a	Motor Armature Inductance [H]	0.0018	0.45
R_a	Motor Armature Resistance [Ω]	0.2	4.5
V	Voltage (D.c) [V]	24	10
η_{rl}	Motor efficiency due to rotational loss	0.87	0.87
η_g	Gear box efficiency	0.85	0.85
W	Weight [Kg]	0.27	1
K	Gain of Potentiometer error (vs/pi) v/rad	24/pi	10/pi

For The Azimuth Servo motor

$$A_m = \frac{\eta K_m K_g}{R_a J_{eq}} = 288.76$$

$$P_m = \frac{B_{eq}}{J_{eq}} + \frac{\eta K_g^2 K_m^2}{R_a J_{eq}} = 40.40$$

Using $k_p=1$ from mat lab Substitute for A_m and P_m

$$\frac{\theta_L(s)}{\theta_i(s)} = \frac{288.76 K_p}{s^2 + 40.40 s + 288.76 K_p}$$

The servomotor transfer function model with gear and position feedback was obtained by

substituting K_p , transfer function model used in this work to control the output angular position θ_L of the DC servomotor with a camera mounted on the shaft. The output response follows the input angle θ_i of azimuth and elevation servomotor as desired for controller tracking of positions. [15].

$$\frac{\theta_L(s)}{\theta_i(s)} = \frac{288.76}{s^2 + 40.40 s + 288.76}$$

For The Elevation Servomotor

$$A_m = \frac{\eta K_m K_g}{R_a J_{eq}} = 42.3$$

$$P_m = \frac{B_{eq}}{J_{eq}} + \frac{\eta K_B^2 K_m^2}{R_a J_{eq}} = 9.6$$

$$\frac{\theta_L(s)}{\theta_i(s)} = \frac{42.3}{s^2 + 9.6s + 42.3}$$

$$\frac{\theta_L(s)}{\theta_i(s)} = \frac{42.3K_p}{s^2 + 9.6s + 42.3K_p}$$

3.2. Adaptive Neuro-Fuzzy (ANFIS) system controller design

Having obtained the servomotor transfer function models for positioning the system vision (camera) it is necessary to design an intelligent non-linear self-tuning controller i.e the ANFIS that controls the servomotors in azimuth and elevation positions to track objects using two dimensional coordinates in a given scene. To obtain the design, the ANFIS must generate change in angular reference position E and derivative position error DE for both the azimuth and elevation axes defined by equations below.

Error (E) = (Desired angular Position–Actual angular Position)

Change in Error (DE) = (Current Error- Previous)

In this model, a first order Sugeno-type fuzzy inference is used for ANFIS and the typical fuzzy rule takes the form of

If x is A_i and y is B_i then,

$Z = f(E, DE)$

Where A_i and B_i are fuzzy sets in the antecedent and $z = f(E, DE)$ is a crisp function in the consequent. The significant of each layer and operation of input output ANFIS structure considered are:

Layer 1: This layer (the fuzzification layer) enables the entry of raw data or crisp inputs from the target system into ANFIS. It is composed of a number of computing nodes whose activation functions are fuzzy logic membership functions, taken here is triangular. Each adaptive node generates the membership grades called fuzzy spaces for the input vectors $i A$, $i = 1, \dots, n$ and $i B$, $i = 1, \dots, n$ where n is the number of membership. Functions of the inputs (E and DE) chosen as $n = 4$. The degree to which the inputs lie within the fuzzy space is given a value normalized between -4 and 4.

$$O_{A,i}^1 = \mu_{A_i}(E), O_{B,i}^1 = \mu_{B_i}(DE), i=1, \dots, n$$

Layer 2: Is the rule layer where each node is fixed. Once the locations of inputs in the fuzzy spaces are identified, the product of the degrees to which the inputs satisfy the membership functions is found. This product is called the firing strength of a rule whose output is given by

$$O_i^2 = W_i = \min(\mu_{A_i}(E) \cdot \mu_{B_i}(DE))$$

In other words, it selects the minimum value of the inputs. In this layer, the total number of Takagi-Sugeno rules used is 16 since the membership function is 4 (4×4) = 16.

Layer 3: This is the normalization layer in which the ratio of each rule's firing strength is calculated with respect to the sum of the firing strengths of all the rules. Each node in this layer is fixed. The node output is the i^{th} input activation level divided by the sum of all the activation levels of the other inputs, as given below

$$O_i^3 = \bar{W}_i = \frac{W_i}{\sum_{i=1}^n W_i}$$

Layer 4: In layer 4, the defuzzification layer, the output of each node is the weighted consequent value. Adaptive node i in this layer calculates the contribution of the i rule towards the overall output, with the following node function

$$O_i^4 = \bar{W}_i Z_i = \bar{W}_i (P_i E + DE q_i + r_i)$$

Layer 5: Layer 5 is the summation layer and its output, which is the sum of all the outputs of layer 4, gives the overall output for the respective inputs within the fuzzy space. The single fixed node in this layer computes the overall output as the sum of each rule's contribution given as

$$O_i^5 = \sum_{i=1}^n \bar{W}_i Z_i = \frac{W_1 Z_1 + W_2 Z_2}{W_1 + W_2}$$

3.2.1 Anfis training for tracking on trajectory with chaotic data

Before ANFIS can be used to predict position of object on a trajectory, training data are presented to the ANFIS. The data presented is for training (estimating) membership function Parameter which fully represents the feature of FIS intending to model. After training, the rules remain so that when new input data is presented to the model, the rules provide a corresponding reasonable output [16]

In this paper, non-linear trajectory movement along x and y is similar to the non-linear movement of object on scene. The optimization technique used is a hybrid learning algorithm that minimizes the error between the ANFIS model and the real system using training data from the target system to generate signals that propagate backwards and forwards and update the parameters. The parameters to be trained are A_i , and B_i of the premise parameters and p_i , q_i and r_i of the consequent parameters. The MATLAB/ANFIS Editor GUI window is used in this work. The ANFIS Editor GUI window includes four distinct areas to support this workflow in realizing this task.

3.2.2 Training procedures

- Loading, Plotting, and Clearing the Data
- Generating or Loading the Initial FIS Structure
- Training the FIS
- Validating the Trained FIS

For generating FIS structure, the triangular membership function (MF) was used for the input variable and output type is linear. The number of MFs for the input variable E and DE is 4 each hence the number of rules is $4 \times 4 = 16$ function for E and DE before training

3.2.3 Simulink Implementation of ANFIS for data training

To implement training procedure for ANFIS desired input/output target control of the system the procedures are as follows;

3.2.4 Loading and training data.

The chaotic data which represent the chaotic trajectory position intended to model are loaded

and checked in ANFIS Matlab GUI shown in figure 5 using the *anfisedit* command window which pops up the ANFIS GUI. Using load data button on ANFIS Matlab GUI, the data representing the model is loaded from disk. The blue small circles represent the load data and the plus sign represent checked data. Fig 6 shows the ANFIS training using the hybrid approach the training error is the difference between the training data output value and the output of the fuzzy inference system. The training error records the root mean square error (RMSE) for each training epoch of the data set. Fig 7 shows how FIS can track train data. Fig 8 and 9 represent the triangular membership function used in training. The rule base for controlling servomotors and the five-layer model structure with 2 inputs and 1 output is depicted in fig 10 and 11.

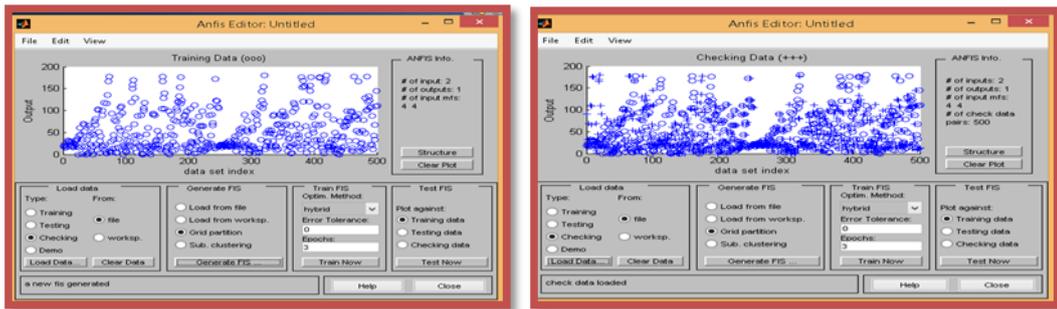


Fig.5 Loading and checking data for training for training in Matlab Anfis Graphical user interface

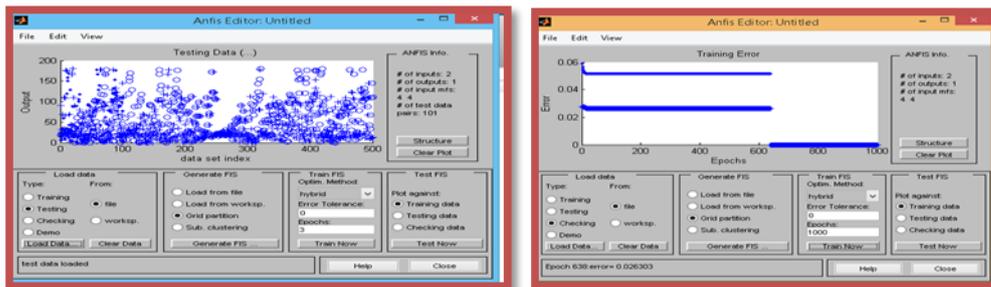


Fig 6 training the output FIS checking the FIS with the minimum error in Graphical user interface

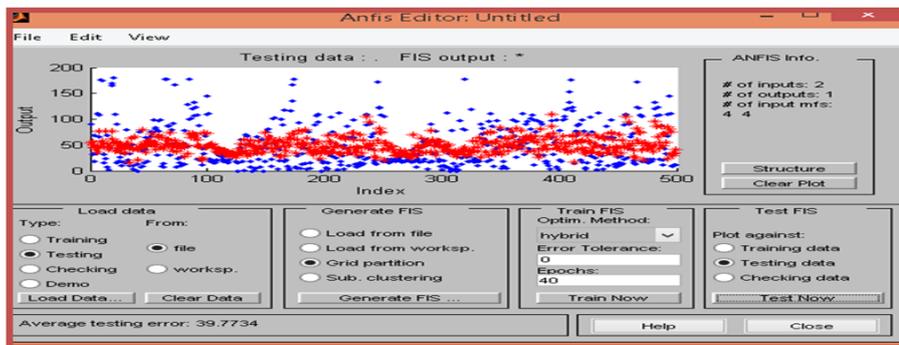


Fig 7 output FIS structure checking trained data

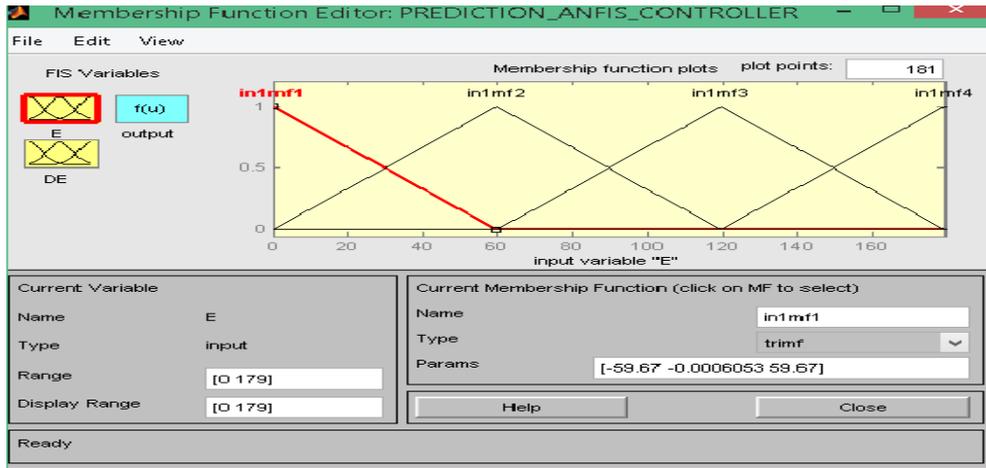


Figure 8 Membership function for training

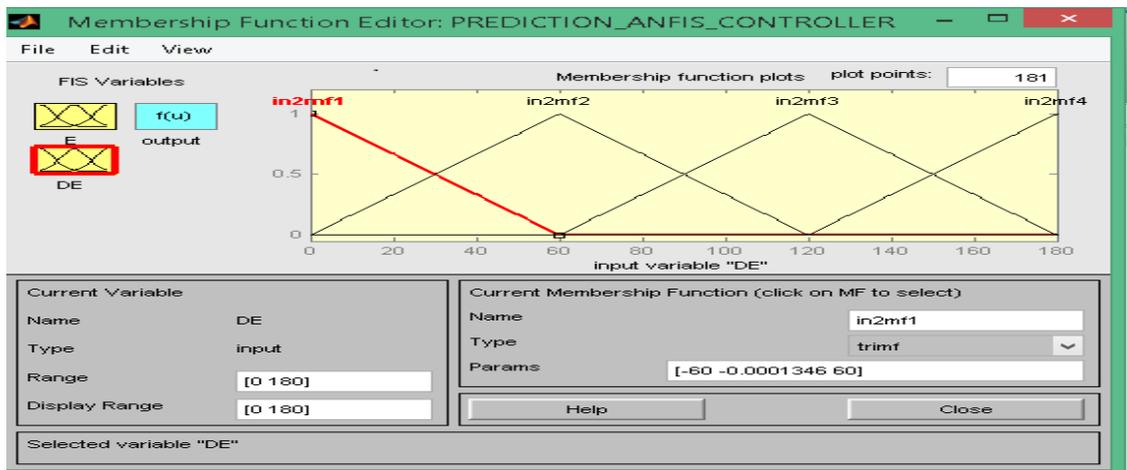


Figure 9 Membership function for training

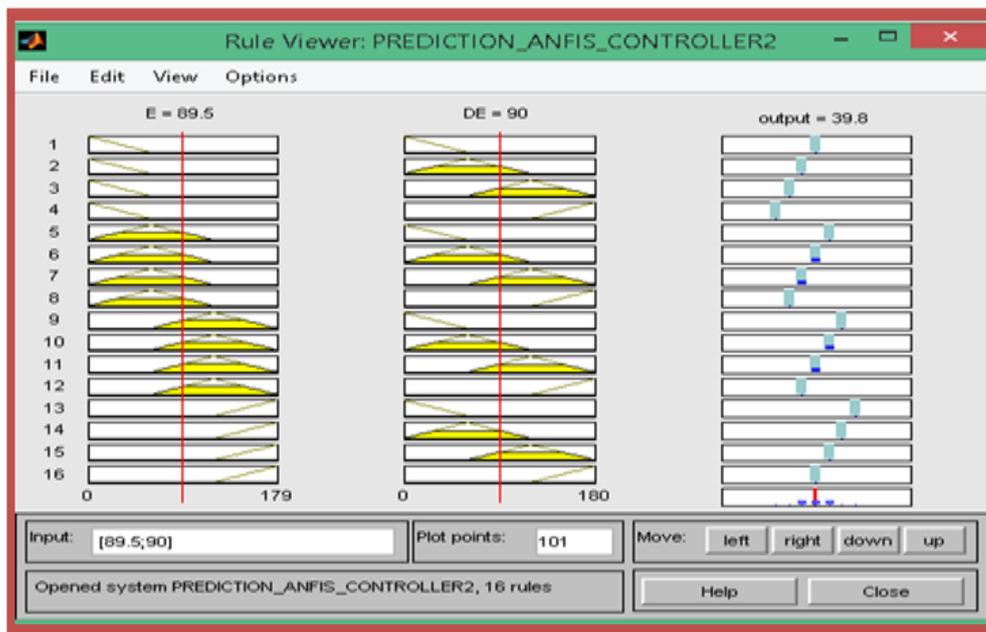


Fig 10 ANFIS control Rule base after training.

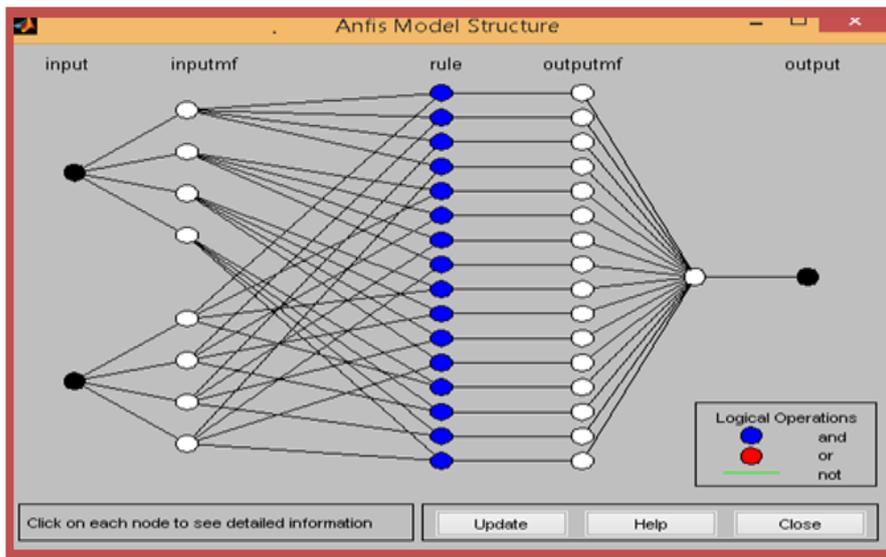


Fig 11 5-Layer ANFIS model structure with 2 inputs & 1 output

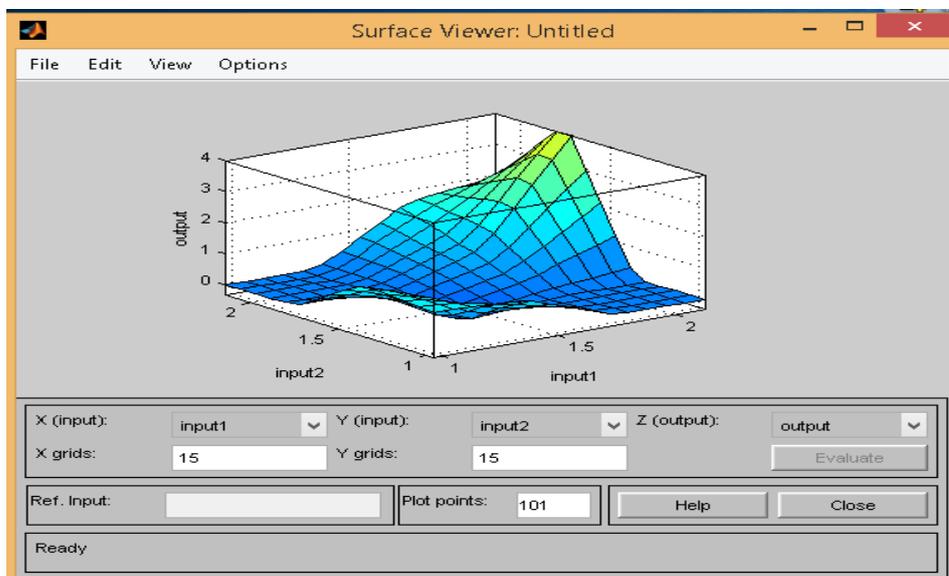


Fig 12 Surface Viewer for two inputs with one output

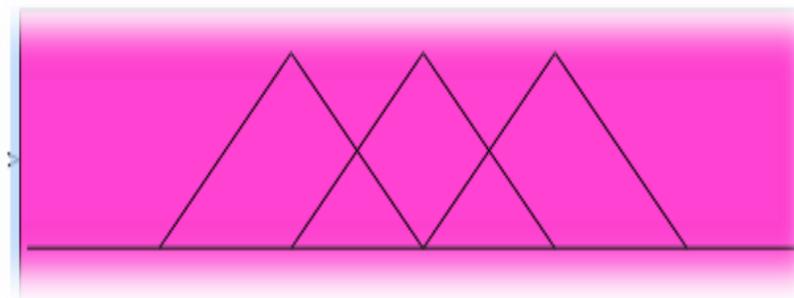


Fig 13 self-turning train FIS embedded (ANFIS) controller

3.3 Servomotors with Anfis Model

Fig 14 and 15 depicts the designed ANFIS model connected to azimuth and elevation servomotors. The connection was simulated to obtain desired performance by the ANFIS

controller. The two models are masked connected to video source with camera mounted shown in Fig 16 to obtain the two dimension track.

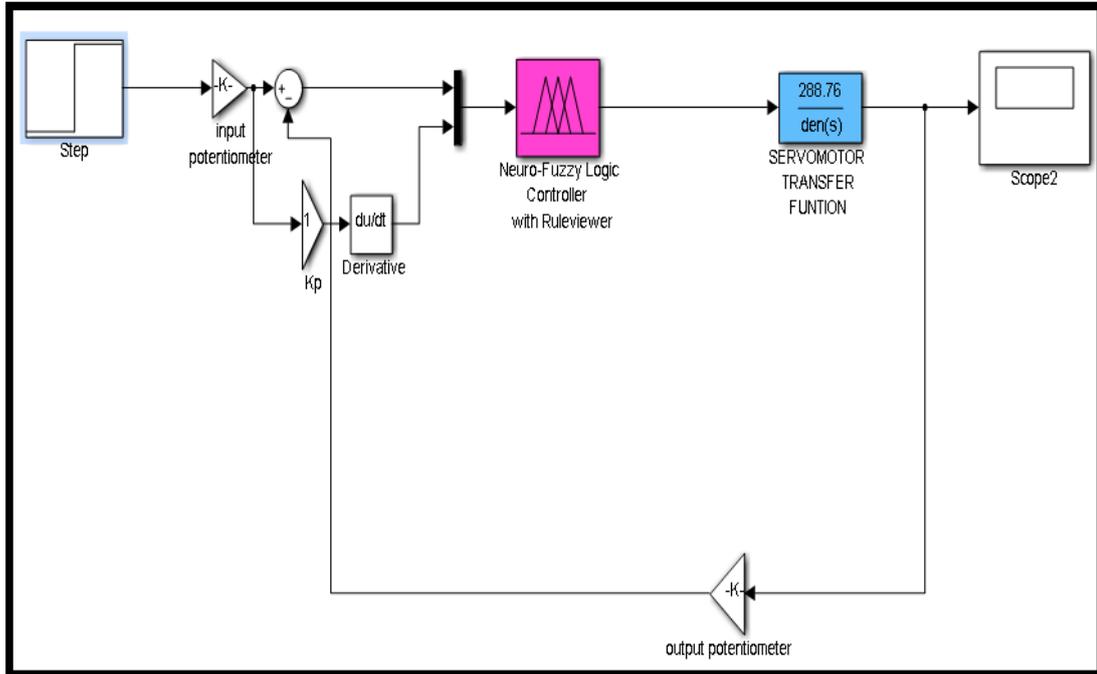


Fig 14 ANFIS model controller with azimuth servomotor with step input

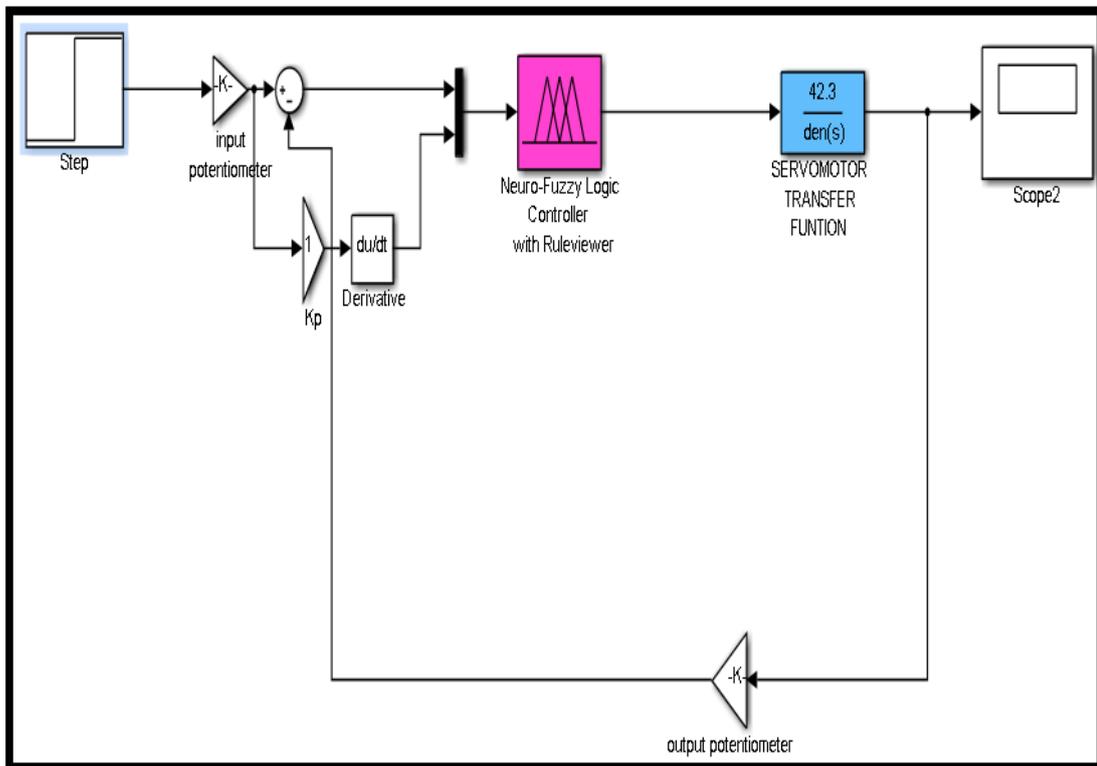


Fig 15 ANFIS model controller with elevation servomotor with step input

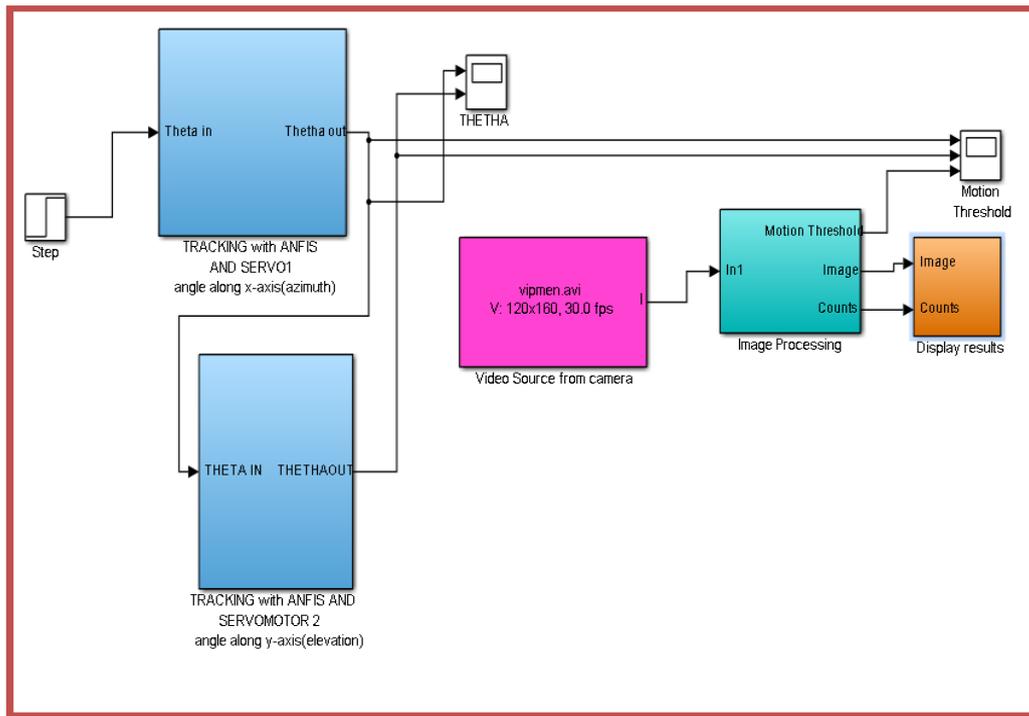


Fig 16 Model Block for object tracking and Motion Prediction

4. Result and Discussions.

The work has been studied by simulation. First, the servo motor transfer functions was simulated using step input for both azimuth and elevation servos without controller, the result is shown in Figure 17. The model servo's without controller were further simulated using angular input signals. With 1degree square Angular positions and sinusoidal signal result figure 18 and 19 were obtained respectively. The result shows that the servos followed the angular positions path or trajectory. The performance of the servomotors to track the

path/trajectory in term of transient, significantly improved with introduction of the model ANFIS controller shown in Fig 20. The complete result for the transient performances of the model is summarize in table 2 this shows the transient result of tracking using servomotors in azimuth and elevation connection with and without the ANFIS controller. The result shows a significant robust performance with the ANFIS controller

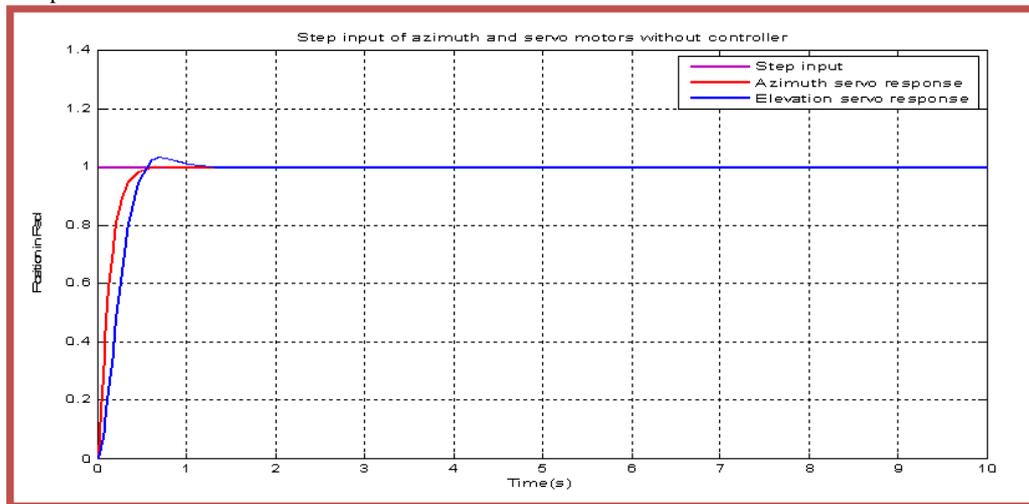


Fig 17 Step response azimuths and elevation servo motor without controller

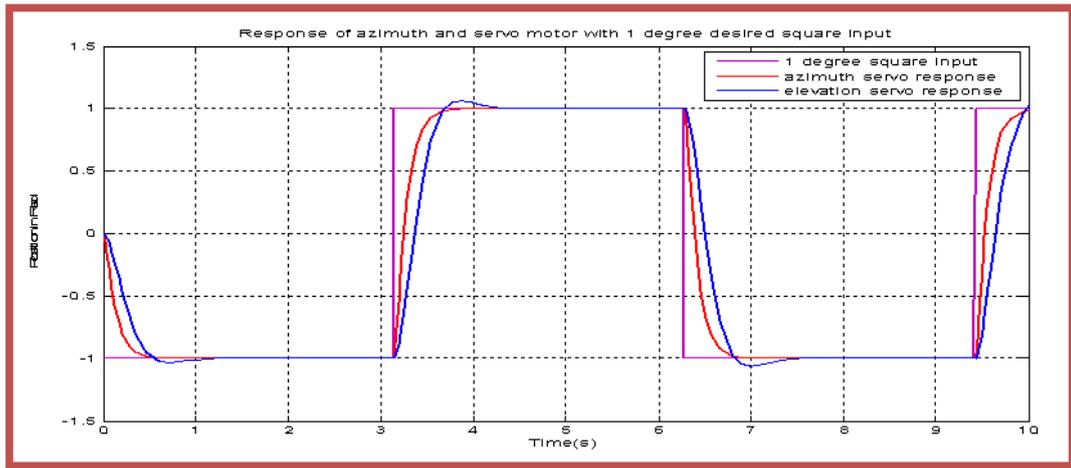


Fig 18 Azimuth and Elevation Servo response with 1 degree square trajectory

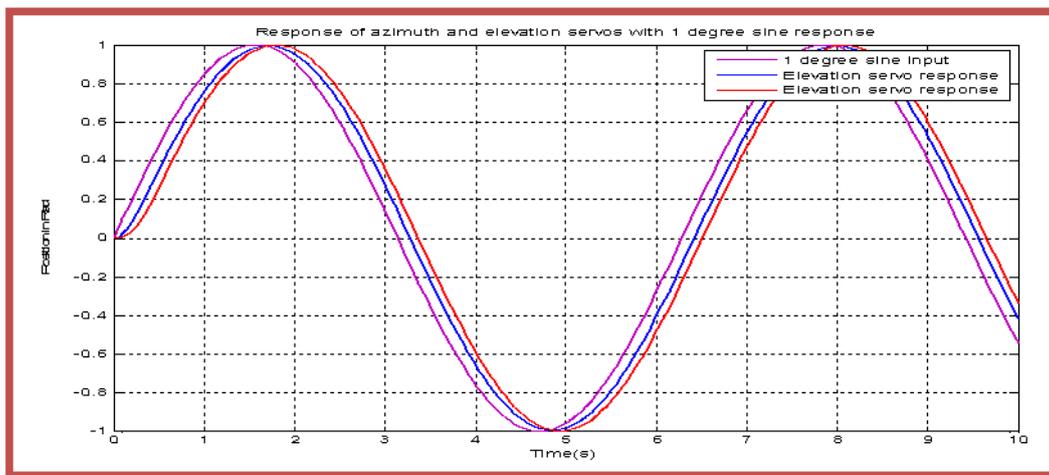


Fig 19 Azimuth and Elevation Servo position response with 1 degree sinusoidal trajectory

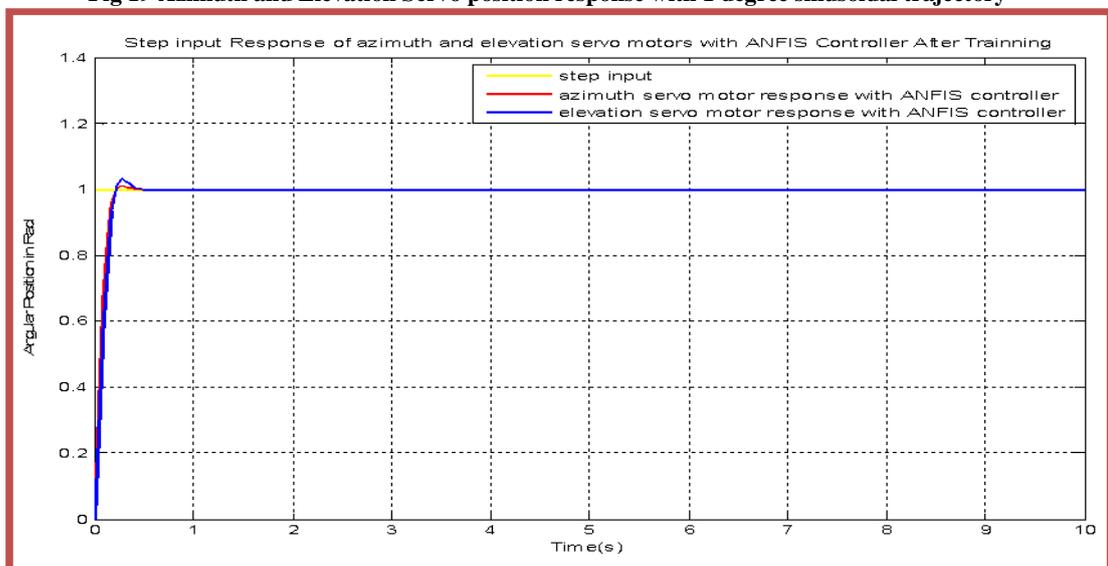


Fig 20 Step responses with azimuth and elevation servo motor position with ANFIS controller

Table 2 Comparison of tracking time response results of azimuth and elevation servo with and without ANFIS controller.

Characteristic specification	Azimuth Servo Without controller	Azimuth Servo With ANFIS controller	Elevation Servo without controller	Elevation Servo with ANFIS controller
Rise time t_r (s)	0.6	0.1	0.6	0.3
settling time t_s (s)	0.6	0.5	1.2	0.5
Peak time t_p (s)	1.0	0.2	0.7	0.2
Delay time t_d (s)	0.3	0.2	0.3	0.1
Maximum Overshoot M_p (%)	1.1	1.0	1.2	1.0

4.1 Result Comparison of Anfis with PID and Fuzzy

To show the performance of ANFIS controller in comparison to the conventional PID and FUZZY on the same servo motor transfer function The result plot for the responses is displayed in scope

plot fig 21 and is summarized in table 3 ANFIS controller output was found the best in terms of faster rise time, settling time and position stabilization.

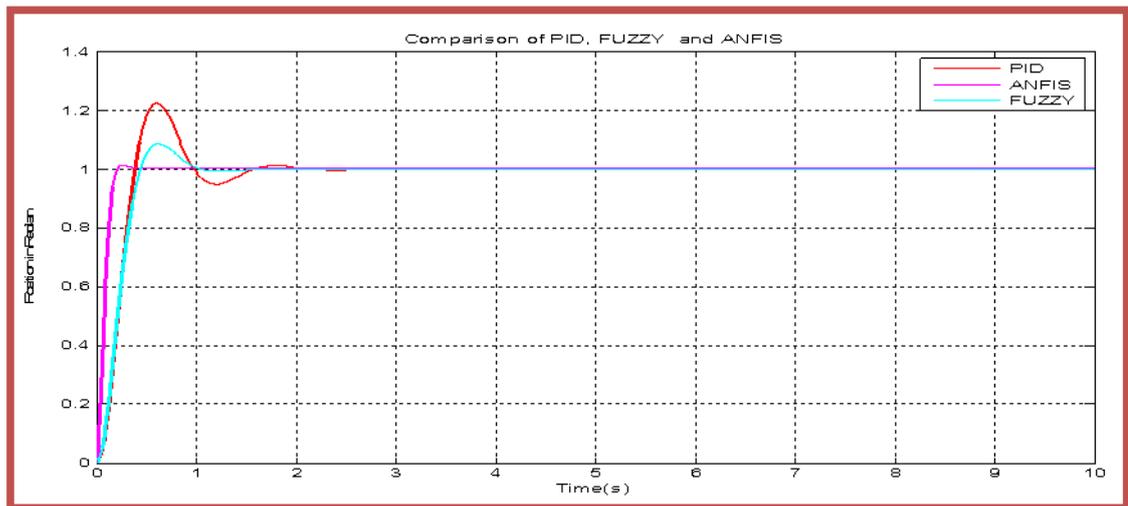


Figure 21 Step input Comparison Plot of PID, FUZZY and ANFIS

Table 3 Step input transient result of PID, FUZZY and ANFIS

Characteristics	PID Controller	FUZZY Controller	ANFIS Controller
t_r (s)	0.6	0.5	0.2
t_s (s)	2.0	1.1	0.5
t_p (s)	0.7	0.6	0.3

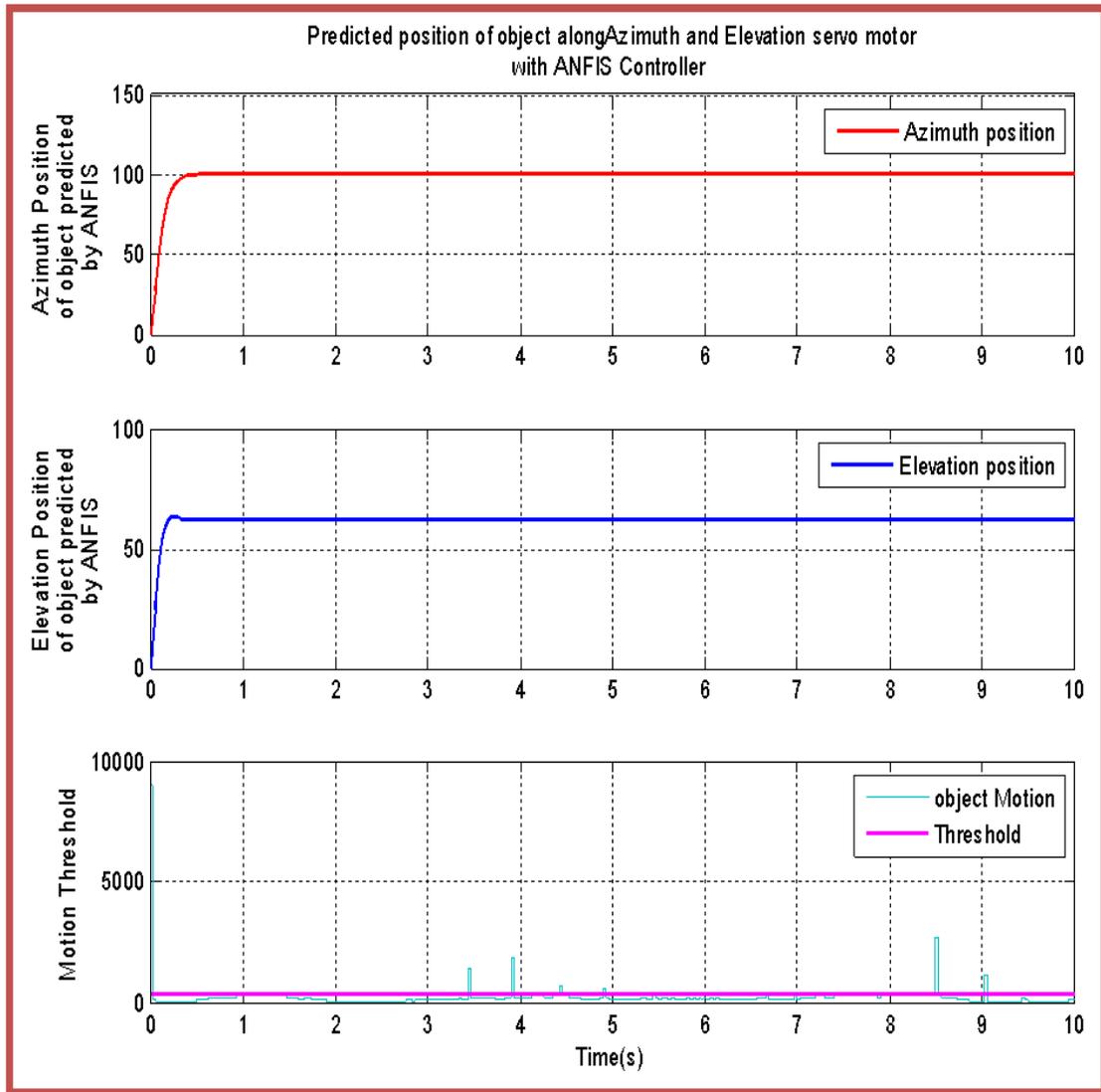


Fig 22 Response of azimuth and elevation positions with ANFIS controller with motion prediction.

Figure 22 shows the separate 2-dimension track position result and motion prediction. The red line in the figure represents the azimuth position and the blue line represents the elevation position. The purple line represents motion threshold during segmentation i.e. all signals above the threshold represent number of motion captured by segmentation.

4.2 Tracking chaotic trajectory with Anfis

The chaotic trajectory result generated using mackley Glass chaotic time series model which is sensitive to initial condition on which the Anfis is to be train for azimuth and elevation trajectory respectively. This chaotic trajectory is necessary since tracking object positions in real life situations

are nonlinear (chaotic).Using chaotic signal input the ANFIS without training were used to track various angular positions displayed shown in Figure 23. However, using ANFIS to learn about this trajectory will produce a better result. The data are trained and validated (checking and testing) on the trajectories. The result plot result Figure 23 shows a trajectory (blue signal) and ANFIS output (red) before training. It can be observed that the ANFIS path which represented by red signal is not able to track the chaotic trajectory represented by blue signal. However Figure 24 shows the plot result of the two signals after ANFIS training. It was noticed that the train ANFIS signal path tracks the trajectory path and was able to follow the chaotic trajectory of object

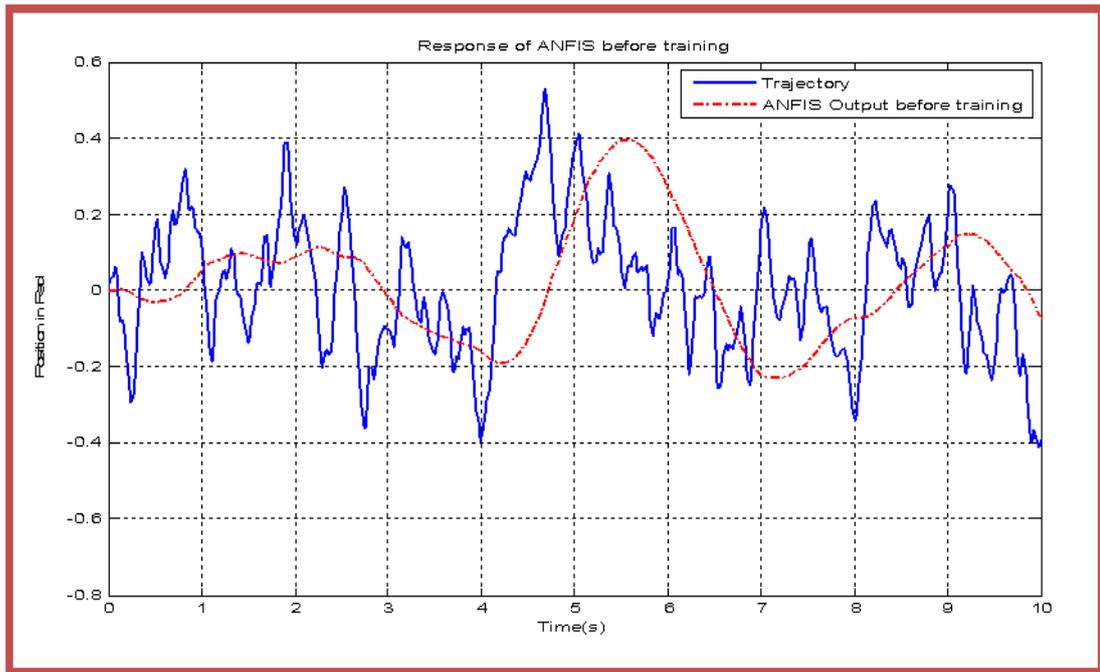


Fig 23 Tracking position along trajectory with servo motor before training

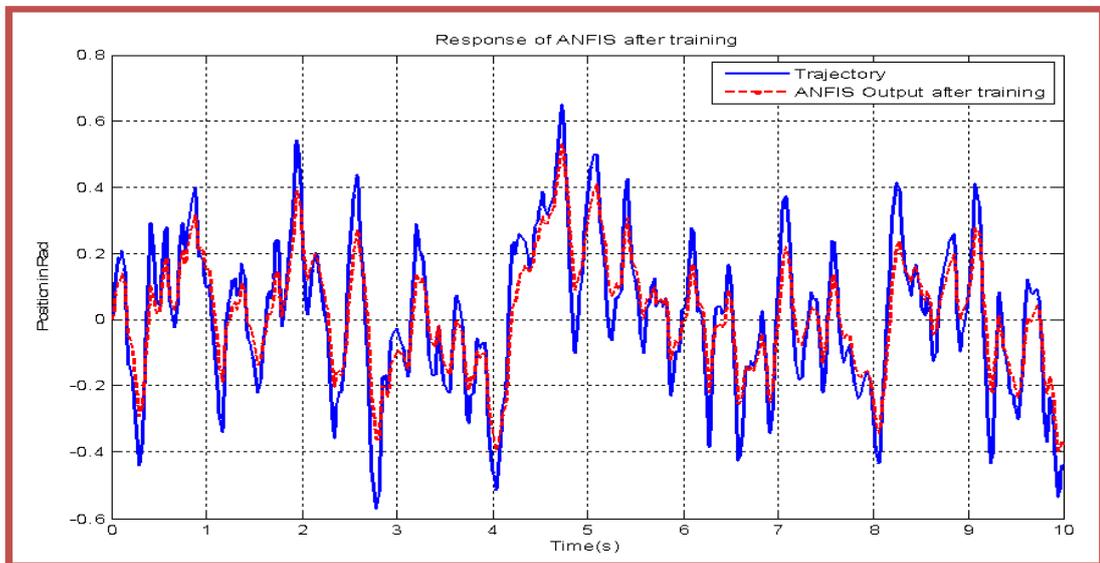


Fig 24 tracking position along trajectory with servo motor after training

5.1 Conclusion

The objective of this paper to design ANFIS Controller for self tuning of DC servomotor has been achieved. ANFIS controller is self tuning the self tuning results a robust control In ANFIS hybrid system. Neural networks are used to tune membership functions of fuzzy systems that are employed as decision-making systems for controlling equipment. The ANFIS controller is efficient with servomotors which have nonlinear characteristics. The prediction by ANFIS compensates the System delay and thus allows precise and fast motion

control viewed through segmentation. ANFIS controller output have faster rise time, settling time, reduced time delay and faster position stabilization in tracking object

5.2 Recommendation

Further work may focus on the hardware implementation application of the developed Neuro-fuzzy system to achieve automatic and real time object tracking with video camera

References

Matlab Image Processing Toolbox User Guide.
(2004).

- [1] Sunitha.M, S. (2013). Real time object Tracking. *International Journal of Emerging Technology and Advance Engineering.*, 3(3), 2250–2459.
- [2] Ramya G, (2014). Real time visual surveillance. *Global Journal of Researches in Engineering*, 14(6).
- [3] Jain A. (1999). Object tracking using Fuzzy Logic for khepera 11 Robot. *Journal of Electrical and Electronics Engineering*, 46(5), 315.
- [4] Lopes. (2011). *Fuzzy Logic based approach for object features.*
- [5] Landge. (2014). Tracking using Background Subtraction. *An international Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering*, 3(7), 45-65
- [6] Ghate & Gawi. (2014). *Object detection using Neural Network. 1(2)*.
- [7] Zhong QU, Quingqing Zhang, T. G. (2012). Moving Object Tracking based on Codebook and Particle Filter. *Internatinal Workshop on Information*
- [8] Yilmaz Alper, J. O. and S. M. (2006). Object Tracking: A survey". *ACM Compt. Surv.*, 1–45.
- [9] Evlampios Apostolidis. (2013). Fast object Re-detection and Localization in video For Spatio- Temporal fragmen. In *International Conference on fast* Molina, José M,
- [10] Jesús García, (2003). Neuro-Fuzzy Technique for image analysis. *IEE Journal of Advance Science Technology*, 5(8), 209–231.
- [11] Jang, S. (2015). Fuzzy Control of data systems for moving vehicle using Robust Cotrollers. *Journal of science Mathematics and Physics* , 36-46.
- [12] Sidney R. Bowes, Fellow, Derrick Holliday. (2004). *New Natural Observer Applied to Speed-Sensor less DC Servo and Induction Motors. IEEE Conference Control.*
- [13] Javiya, K. a. (2016). Comparisons of Different Controller for Position Tracking of DC Servo Motor. *International Journal of Advanced Research in Electrical Engineering*, Vol. 5 (Issue 2), 966-967.
- [14] Bolton, W. (*Experiment Note,1999*). pp. 1-10.
- [15] Jane,(Handbook of Servo Control,2014)
- [16] Lofti A. Zadeh and Berkeley, C. (1995). *Fuzzy Logic Toolbox User's Guide.*